

# Another Remarkable Work

GRAPHIC, SOUND, COMPILER

**X1ハイテックファイル**

渡辺英行, 高橋秀樹共著









GRAPHIC, SOUND, COMPILER



# ハイテックファイル

渡辺英行, 高橋秀樹 共著

MIA







## まえがき

本書は、『マシン語プログラミング入門』、『X1リファレンスノット』に続く第3弾として、また本シリーズの最終作として、これら既刊2冊のアンケート葉書のなかで、もっとも要望が多かったアイテムを取り上げました。内容は、ゲーム制作のためのキャラクタ・パターン表示、CIRCLEやLINEなどのグラフィックス、音声合成を含むPSG活用法、開発ツールとしてのコンパイラです。

これらはもちろん、すべてマシン語で記述しますが、特にアルゴリズムが重要なグラフィックスについては、BASICでも記述し、アルゴリズムを明確にしています。

コンパイラは、オブジェクト効率を重視したもので、X1のようにPCGなどの機能が充実しているマシンでは、リアルタイム・ゲームの記述にも十分使えます。

BASICとマシン語以外の言語として、初めて使う方には多少とまどいがあるかもしれませんが、ローカル変数が使え点や豊富な制御文などの構造化言語の雰囲気味わってください。

1985年 8月 著者



## ■ 本書を読む前に ■

本書で使用したアセンブラは、(株)アスキーの『DUAD-X1』です。しかし、本シリーズの『マシン語プログラミング入門』に掲載されたアセンブラを使っている方でもアセンブルできるように配慮しているので問題はないでしょう。

1 章と 2 章では、グラフィック RAM を使いますが、両アセンブラとも、グラフィック表示を禁止しています。ですから、プログラムを実行するときは BASIC から実行するか、つぎのプログラムを実行してパレットを初期設定してから行ってください。

ld	bc, 1000h	; パレット B
ld	a, 0aah	
out	(c), a	
inc	b	; パレット R
ld	a, 0cch	
out	(c), a	
inc	b	; パレット G
ld	a, 0f0h	
out	(c), a	

また、グラフィックのモードを変えるときは、'width 80' (098ch) または 'width 40' (0998h) をそれぞれコールしてください。

なお、X1 turbo も含む全 X1 シリーズに対応するため BASIC は『CZ-8FB01』と『CZ-8CB01』を対象とします。





# CONTENTS





## 第1章 ゲーム制作のノウハウ—————1

### 1-1 基礎編—————2

#### 1-1-1 テキスト画面とアトリビュート—————2

#### 1-1-2 グラフィック画面—————6

### 1-2 パターンの移動—————11

#### 1-2-1 キー入力—————11

#### 1-2-2 基本的移動—————11

#### 1-2-3 大きなパターンの移動—————15

#### 1-2-4 横4ドット単位の移動—————20

### 1-3 応用テクニック—————26

#### 1-3-1 重ね合わせ—————26

#### 1-3-2 PCGによるスクロール—————38

## 第2章 グラフィックス—————41

### 2-1 PSET—————42

#### 2-1-1 アルゴリズム—————42

#### 2-1-2 プログラム—————43

### 2-2 LINEとCIRCLE—————48

#### 2-2-1 LINE—————48

#### 2-2-2 CIRCLE—————52



2-3 PAINT	59
-----------	----

## 第3章 サウンド活用テクニック 73

3-1 効果音	74
---------	----

3-1-1 基本サブルーチン	74
----------------	----

3-1-2 ノイズ・ジェネレータの活用	76
---------------------	----

3-1-3 特殊効果音	79
-------------	----

3-2 音声合成	86
----------	----

## 第4章 Stellarコンパイラ 89

4-1 Stellarコンパイラの概要	90
---------------------	----

4-1-1 コマンドの説明	90
---------------	----

4-1-2 ファンクション・キー	91
------------------	----

4-2 エディタ	93
----------	----

4-2-1 カーソル移動	95
--------------	----

4-2-2 インサート&デリート	96
------------------	----

4-2-3 ファインド&リプレイス	97
-------------------	----

4-2-4 ブロック・オペレーション	99
--------------------	----

4-2-5 その他	102
-----------	-----



4-3 モニタ	103
4-3-1 Rコマンド	103
4-3-2 Gコマンド	103
4-3-3 Xコマンド	104
4-3-4 Hコマンド	104
4-4 コンパイラ	106
4-4-1 Stellarの構文と文法	106
4-4-2 Stellarエラー メッセージ	146
4-4-3 コンパイラの使い方	149
4-4-4 マシン語とのインターフェイス	155
4-4-5 ライブラリの使い方	159
4-4-6 コンパイラの全リスト	173

■APPENDIX	195
-----------	-----



# 第1章 ゲーム制作のノウハウ

---

## 1-1 基礎編

### 1-2 パターンの移動

### 1-3 応用テクニック

---

本章では、PCGによる文字表示やグラフィック・パターンの移動、それに伴う特殊な移動や背景との重ね合わせ、PCGによるスクロールなど、リアルタイム・ゲームに必要な表示のテクニックを順次説明していきます。なお、簡単のためグラフィックのモードは320×200ドットとします。また、サンプルのプログラムはBASICから実行してください。



# 1-1

# 基礎編

キャラクタ・パターンの表示はゲームをつくるうえで重要な部分です。本編では、文字や PCG およびグラフィック・パターンの表示方法について解説していきます。

## 1-1-1 テキスト画面とアトリビュート

テキスト画面には、ROM（キャラクタ・ジェネレータ ROM）に内蔵されている文字フォントや PCG（Programable Character Generator）に定義されたパターンが表示されます。この画面には、専用の VRAM（テキスト VRAM）が I / O ポートの 3000 H 番地から 37 FFH 番地までに割り当てられていて、画面の位置に対応する番地に、ASCII コードを書き込めばそのコードの文字が表示されます。また、この文字に対して、1 文字単位に、キャラクタ・ジェネレータ ROM、PCG の選択、文字のサイズ、反転、点滅、色指定をするアトリビュート VRAM が別に設けられています。

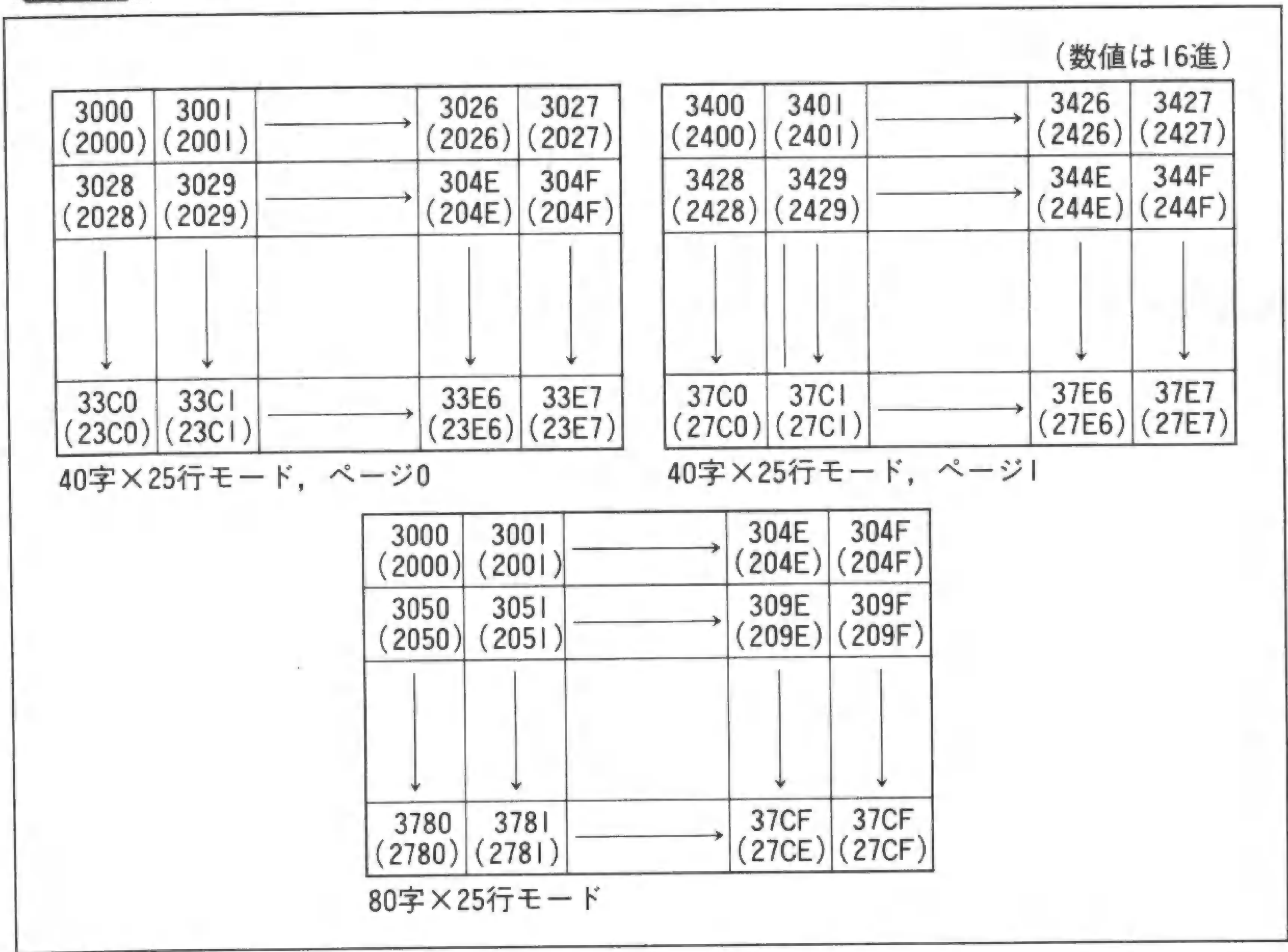
X 1 には、表示モードが 2 種類（4 0 × 2 5，8 0 × 2 5）あり、表示位置と I / O アドレスは図 1-1 のように対応しています。図のカッコ内の数値はアトリビュートのアドレスです。

アトリビュート・アドレスは、テキスト・アドレス-1000 H で求められますが、これはたとえば bc レジスタがテキスト・アドレスを示している場合、アドレスの上位 8 ビットが格納されている b レジスタの第 4 ビットをクリアすることで簡単に求めることができます。

例	ld	bc, nn (nnは、テキスト・アドレス)
	res	4, b



図1-1 テキスト画面の表示位置とアトリビュート



アトリビュートは、アトリビュート VRAM に1バイトのデータを書き込んで指定します。アトリビュートの各ビットは図1-2のような意味があります。

表示モード別にテキスト画面のアドレスは、つぎの式で計算できます。

- 40文字×25行 (0 ページ)  
アドレス = &H 3000 + X座標 + Y座標 × 40
- 40文字×25行 (1 ページ)  
アドレス = &H 3400 + X座標 + Y座標 × 40
- 80文字×25行  
アドレス = &H 3000 + X座標 + Y座標 × 80

リスト 1-1 は、上記の式のアセンブル・リストです。  
リスト 1-2 は、実際に文字を表示するプログラムです。このプログラムは、'xpos'に X 座標, 'ypos'に Y 座標, 'letter'



に ASCII コード，‘code’にアトリビュート・コードを入れて  
コールするものです。ここでは例として PCG の ASCII コ  
ード 41 H にあたる文字を画面左上に表示し，さらに，その  
右にアトリビュート・コードを変えて“A”と表示しています。

図1-2 アトリビュートのビット内容

ビット番号	説 明																																							
0 ~ 2	キャラクターの色を指定します。カラー 8 色の表示が可能。 <table><tr><th colspan="3">ビット</th><th rowspan="2">指 定 色</th></tr><tr><th>2</th><th>1</th><th>0</th></tr><tr><td>0</td><td>0</td><td>0</td><td>黒</td></tr><tr><td>0</td><td>0</td><td>1</td><td>青</td></tr><tr><td>0</td><td>1</td><td>0</td><td>赤</td></tr><tr><td>0</td><td>1</td><td>1</td><td>赤紫(マゼンタ)</td></tr><tr><td>1</td><td>0</td><td>0</td><td>緑</td></tr><tr><td>1</td><td>0</td><td>1</td><td>水 色(シアン)</td></tr><tr><td>1</td><td>1</td><td>0</td><td>黄</td></tr><tr><td>1</td><td>1</td><td>1</td><td>白</td></tr></table>	ビット			指 定 色	2	1	0	0	0	0	黒	0	0	1	青	0	1	0	赤	0	1	1	赤紫(マゼンタ)	1	0	0	緑	1	0	1	水 色(シアン)	1	1	0	黄	1	1	1	白
ビット			指 定 色																																					
2	1	0																																						
0	0	0	黒																																					
0	0	1	青																																					
0	1	0	赤																																					
0	1	1	赤紫(マゼンタ)																																					
1	0	0	緑																																					
1	0	1	水 色(シアン)																																					
1	1	0	黄																																					
1	1	1	白																																					
3	“1”→ビット0~2で指定したキャラクター・カラーの補色表示(反転)を行なう。																																							
4	“1”→キャラクターを約0.5秒周期で点滅させます。																																							
5	文字の表示モードを指定します。 <table><tr><th>ビット内容</th><th>設 定 モ ー ド</th></tr><tr><td>0</td><td>標準文字モード (CG ROM)</td></tr><tr><td>1</td><td>ユーザー文字モード (PCG)</td></tr></table>	ビット内容	設 定 モ ー ド	0	標準文字モード (CG ROM)	1	ユーザー文字モード (PCG)																																	
ビット内容	設 定 モ ー ド																																							
0	標準文字モード (CG ROM)																																							
1	ユーザー文字モード (PCG)																																							
6 ~ 7	キャラクターの大きさを変えます。 <table><tr><th colspan="2">ビット</th><th rowspan="2">機 能</th></tr><tr><th>7</th><th>6</th></tr><tr><td>0</td><td>0</td><td>ノーマル文字</td></tr><tr><td>0</td><td>1</td><td>垂直 2 倍文字</td></tr><tr><td>1</td><td>0</td><td>水平 2 倍文字</td></tr><tr><td>1</td><td>1</td><td>垂直・水平 2 倍文字</td></tr></table>	ビット		機 能	7	6	0	0	ノーマル文字	0	1	垂直 2 倍文字	1	0	水平 2 倍文字	1	1	垂直・水平 2 倍文字																						
ビット		機 能																																						
7	6																																							
0	0	ノーマル文字																																						
0	1	垂直 2 倍文字																																						
1	0	水平 2 倍文字																																						
1	1	垂直・水平 2 倍文字																																						



リスト1-1

```
1      ;
2      ;      --- Text address ---
3      ;
4      ;
5 C000      org      0C000H
6      ;
7      ;
8 C000 210030 tadr: ld      hl,3000H      ;Base address of text ram
9 C003 3A1EC0      ld      a,(ypos)      ;y position in a
10 C006 FE00      cp      0              ;y=0 ?
11 C008 2807      jr      z,skip          ;If y=0 then skip
12 C00A 012800      ld      bc,28H        ;x colome in bc (40)
13 C00D 09      loop: add      hl,bc
14 C00E 3D      dec      a
15 C00F 28FC      jr      z,loop
16 C011 ED5B1DC0 skip: ld      de,(xpos)      ;x position in de
17 C015 19      add      hl,de
18 C016 44      ld      b,h
19 C017 4D      ld      c,l              ;Address in bc
20 C018 ED431FC0      ld      (address),bc
21 C01C C9      ret
22      ;
23 C01D 00      xpos: defb      0
24 C01E 00      ypos: defb      0
25      ;
26 C01F      address: defb      2
```

リスト1-2

```
1      ;
2      ;      --- Print routine (PCG) ---
3      ;
4      ;
5 C000      org      0C000H
6      ;
7      ;
8 C000 CD21C0      call      tadr          ;Address of text ram in bc
9      ;
10      ;      ---- Print routine (address of text ram in bc) ----
11      ;
12 C003 3A3CC0      ld      a,(letter)      ;Ascii code of letter in a
13 C006 ED79      out      (c),a            ;Print
14 C008 CBA0      res      4,b              ;Address of attribute ram
15 C00A 3A3DC0      ld      a,(code)        ;Code of attribute
16 C00D ED79      out      (c),a
17 C00F CBE0      set      4,b
18      ;
19 C011 03      inc      bc                  ;bc=bc+1
20 C012 3A3CC0      ld      a,(letter)
21 C015 ED79      out      (c),a
22 C017 CBA0      res      4,b
23 C019 3A3EC0      ld      a,(code+1)
24 C01C ED79      out      (c),a
25 C01E CBE0      set      4,b
26 C020 C9      ret
27      ;
28      ;      ---- calculate address of text ram ----
29      ;
30 C021 210030 tadr: ld      hl,3000H      ;Base address of text ram
31 C024 3A3BC0      ld      a,(ypos)      ;y position in a
32 C027 FE00      cp      0              ;y=0 ?
33 C029 2807      jr      z,skip          ;If -y=0 then skip
34 C02B 012800      ld      bc,28H        ;x colome in bc (40)
35 C02E 09      loop: add      hl,bc
36 C02F 3D      dec      a
37 C030 28FC      jr      z,loop
38 C032 ED5B3AC0 skip: ld      de,(xpos)      ;x position in de
39 C036 19      add      hl,de
```

つづく



リスト1-2 つづき

40	C037	44		ld	b,h	
41	C038	4D		ld	c,l	;Address in bc
42	C039	C9		ret		
43			:			
44	C03A	00	xpos:	defb	0	
45	C03B	00	ypos:	defb	0	
46			:			
47	C03C	41	letter:	defb	41H	
48			:			
49	C03D	27	code:	defb	00100111B	;27h(normal,ramcg,white)
50	C03E	07		defb	00000111B	;07h(normal,romcg,white)

## 1-1-2 グラフィック画面

グラフィック画面には、専用の VRAM が I / O ポート 4000 H 番地から FFFFH 番地までに割り当てられています。構成は図 1-3 のとおりです。

BLUE のアドレスに 4000 H を加えた値が RED のアドレスになり、さらに、RED のアドレスに、4000 H を加えた値が、GREEN のアドレスになります。

画面のキャラクタ座標から、グラフィック RAM のアドレスを求めるプログラムがリスト 1-3 です。'xpos'に X 座標、'ypos'に Y 座標を入れてコールします。

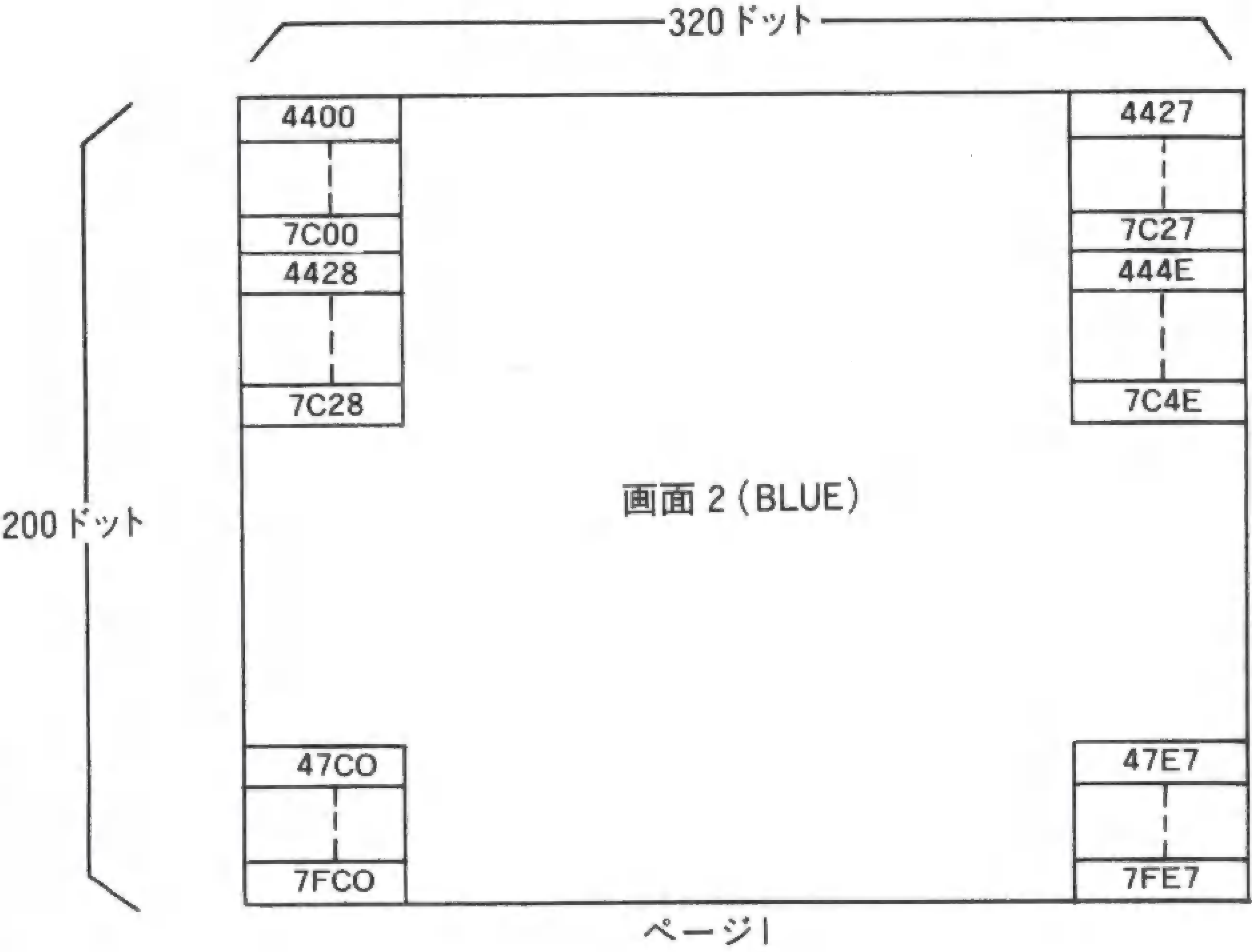
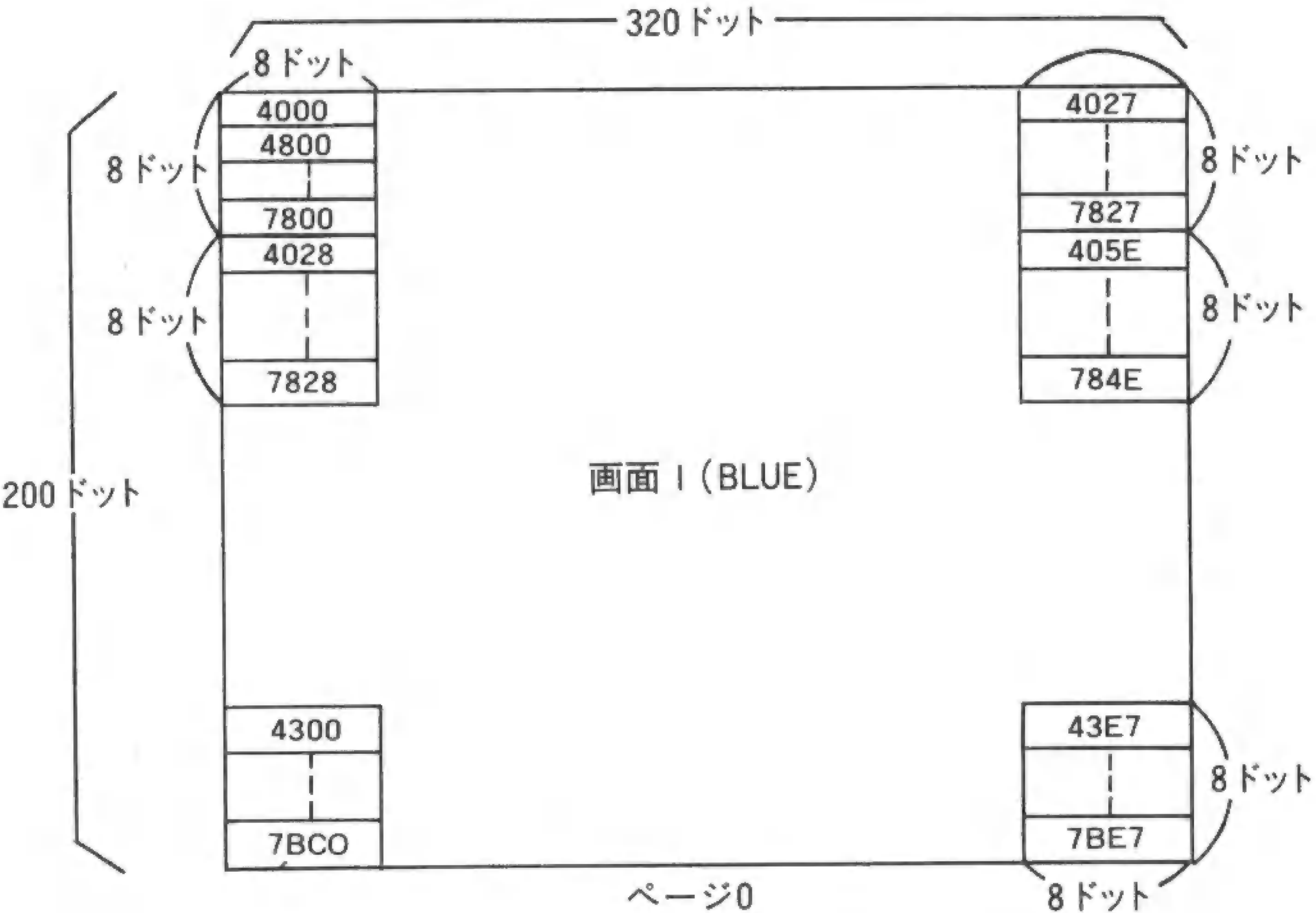
グラフィック・アドレスは、すべて計算で求めてもよいのですが、この方法は後章にゆずるとして、ここでは、画面の座標をあらかじめデータとして持つことで処理します。これは特に、高速にグラフィック・パターンを動かしたい場合などに有効です。

リスト 1-4 は、16 ドット×16 ドットのグラフィック・パターンを表示するプログラムです。



図1-3

グラフィックVRAM(BLUE)アドレスと表示位置との関係

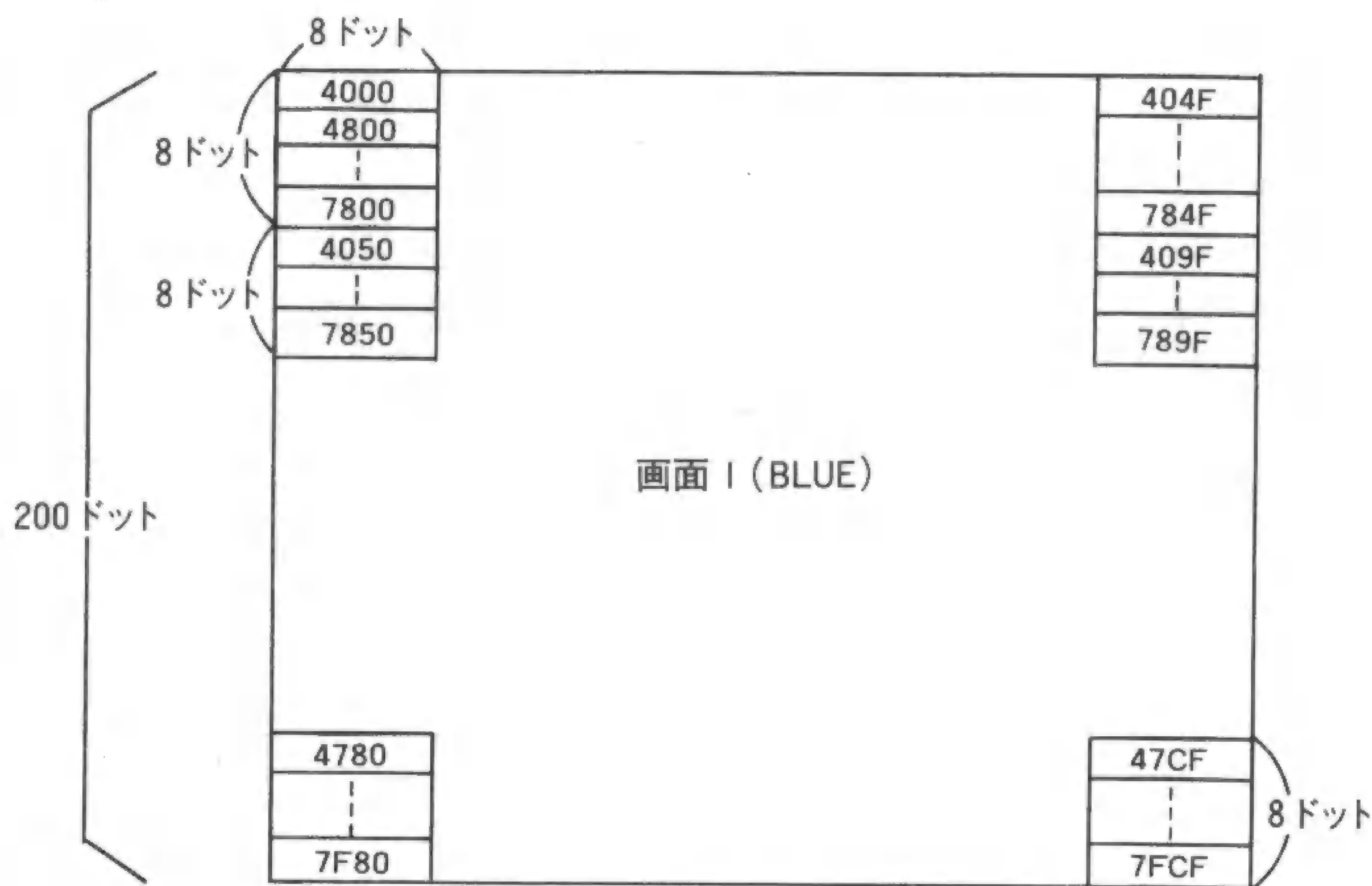


(a) 320×200ドット画面 (2画面)

つづく



図1-3 つづき



(b) 640×200ドット画面 (1画面)

リスト1-3

```

1      ;
2      ;      --- address ---
3      ;
4 C000      org      0C000H
5      ;
6 C000 0117C0      ld      bc,table
7 C003 2A4BC0      ld      hl,(ypos)      ;y position/8
8 C006 29          add     hl,hl
9 C007 09          add     hl,bc
10 C008 5E         ld      e,(hl)
11 C009 23         inc     hl
12 C00A 56         ld      d,(hl)
13 C00B 2A49C0     ld      hl,(xpos)      ;x position/8
14 C00E 19         add     hl,de
15 C00F 110040     ld      de,4000H      ;base address of vram (blue)
16 C012 19         add     hl,de      ;address
17 C013 2200D0     ld      (0D000H),hl      ;save address to d000h
18 C016 C9         ret
19      ;
20 C017 00002800   table: defw    0,40,80,120,160,200,240,280,320,360
    C01B 50007800
    C01F A000C800
    C023 F0001801
    C027 40016801
21 C02B 9001B801   defw    400,440,480,520,560,600,640,680,720
    C02F E0010802
    C033 30025802
    C037 8002A802
    C03B D002
22 C03D F8022003   defw    760,800,840,880,920,960
    C041 48037003
    C045 9803C003
23      ;
24 C049 0000      xpos:   defw    0
25 C04B 0000      ypos:   defw    0

```



リストI-4

```

1      ;
2      ; ---- Grafic pattern print routine ----
3      ;
4      ;
5 C000      org      0C000H
6      ;
7      ; ---- Calculate address of vram ----
8      ;
9 C000 3A6CC0      ld      a,(ypos)      ;y position / 8 in a
10 C003 87          add     a,a          ;Double index
11 C004 6F          ld      l,a          ;Make index (8bit) in hl
12 C005 2600        ld      h,0
13 C007 116DC0      ld      de,table     ;address of table in de
14 C00A 19          add     hl,de        ;Table+index in hl
15 C00B 4E          ld      c,(hl)       ;Get lsb in c
16 C00C 23          inc     hl
17 C00D 46          ld      b,(hl)       ;Get msb in b
18 C00E 2A6AC0      ld      hl,(xpos)    ;x position / 8 in hl
19 C011 09          add     hl,bc
20 C012 CBF4        set     6,h          ;Base address of vram (blue)
21 C014 44          ld      b,h
22 C015 4D          ld      c,l
23      ;
24      ; ---- display routine ----
25      ;
26 C016 219FC0      ld      hl,pdata     ;Address of pattern data in hl
27 C019 3E03        ld      a,3          ;Set counter
28 C01B F5          push    af           ;Save counter
29 C01C C5          loop1: push    bc     ;Save address of vram
30 C01D E5          push    hl           ;Save address of pattern data
31 C01E 111000      ld      de,0010H     ;Pdata+16 in de
32 C021 19          add     hl,de        ;hl+16 in hl
33 C022 EB          ex      de,hl
34 C023 212800      ld      hl,28H       ;hl+x colome in hl
35 C026 09          add     hl,bc
36 C027 2249C0      ld      (td+1),hl
37 C02A E1          pop     hl
38      ;
39 C02B 3E02        ld      a,2          ;Set counter
40 C02D F5          loop2: push    af     ;Save counter
41 C02E 3E08        ld      a,8          ;Set counter
42 C030 F5          loop3: push    af     ;Save counter
43 C031 7E          ld      a,(hl)       ;Pattern data in a
44 C032 ED79        out     (c),a        ;Print
45 C034 03          inc     bc           ;Increment address of vram
46 C035 1A          ld      a,(de)       ;Pattern data in a
47 C036 ED79        out     (c),a
48 C038 0B          dec     bc           ;Decremant address of vram
49 C039 23          inc     hl           ;Increment address of pattern data
50 C03A 13          inc     de           ;Increment address of pattern data
51 C03B E5          push    hl
52 C03C 210008      ld      hl,0800H     ;bc+800H (1 line)
53 C03F 09          add     hl,bc
54 C040 44          ld      b,h
55 C041 4D          ld      c,l
56 C042 E1          pop     hl
57 C043 F1          pop     af           ;Reload counter
58 C044 3D          dec     a           ;Decrement counter
59 C045 20E9        jr      nz,loop3     ;If a<>0 then loop3
60 C047 F1          pop     af           ;Reload counter
61 C048 010000      td: ld      bc,0000H
62 C04B 3D          dec     a           ;Decremant counter
63 C04C 20DF        jr      nz,loop2
64 C04E C1          pop     bc           ;Reload address of vram
65 C04F E5          push    hl
66 C050 2A49C0      ld      hl,(td+1)
67 C053 7C          ld      a,h
68 C054 FEC0        cp      0C0H         ;Is vram green ?
69 C056 D267C0      jp      nc,quit      ;If green then quit
70 C059 210040      ld      hl,4000H
71 C05C 09          add     hl,bc
72 C05D 44          ld      b,h
73 C05E 4D          ld      c,l
74 C05F E1          pop     hl
75 C060 111000      ld      de,10H
76 C063 19          add     hl,de
77 C064 C31CC0      jp      loop1

```



リスト-4

つづき

```

78      ;
79 C067 E1      quit:    pop    hl
80 C068 F1      pop    af
81 C069 C9      ret
82      ;
83 C06A 0000     xpos:    defw    0          ;x position
84 C06C 00      ypos:    defb    0          ;y position
85      ;
86 C06D 00002800 table:    defw    0,40,80,120,160,200,240,280,320,360
      C071 50007800
      C075 A000C800
      C079 F0001801
      C07D 40016801
87 C081 9001B801      defw    400,440,480,520,560,600,640,680,720
      C085 E0010802
      C089 30025802
      C08D 8002A802
      C091 D002
88 C093 F8022003      defw    760,800,840,880,920,960
      C097 48037003
      C09B 9803C003
89      ;
90 C09F 000A3F3F pdata:    defb    00H,0AH,3FH,3FH,3FH,3DH,3DH,3FH      ;blue
      C0A3 3F3D3D3F
91 C0A7 3F3C3C3F      defb    3FH,3CH,3CH,3FH,3FH,3FH,3FH,00H
      C0AB 3F3F3F00
92 C0AF 00A0FCFC      defb    00H,0A0H,0FCH,0FCH,0FCH,0BCH,0BCH,0FCH
      C0B3 FCBCBCFC
93 C0B7 FC3C3CFC      defb    0FCH,3CH,3CH,0FCH,0FCH,0FCH,0FCH,00H
      C0BB FCFCFC00
94      ;
95 C0BF 073066CE      defb    07H,30H,66H,0CEH,0CEH,4CH,0CH,0EH      ;red
      C0C3 CE4C0C0E
96 C0C7 46DBDB80      defb    46H,0DBH,0DBH,80H,00H,00H,00H,0F9H
      C0CB 000000F9
97 C0CF E00C6673      defb    0E0H,0CH,66H,73H,73H,32H,30H,70H
      C0D3 73323070
98 C0D7 62DBDB01      defb    62H,0DBH,0DBH,01H,00H,00H,00H,9FH
      C0DB 0000009F
99      ;
100 C0DF 073F66CE      defb    07H,3FH,66H,0CEH,0CEH,4CH,0CH,0EH      ;green
      C0E3 CE4C0C0E
101 C0E7 46C3C380      defb    46H,0C3H,0C3H,80H,00H,00H,00H,0F8H
      C0EB 000000F8
102 C0EF E0FC6673      defb    0E0H,0FCH,66H,73H,73H,32H,30H,70H
      C0F3 73323070
103 C0F7 62C3C301      defb    62H,0C3H,0C3H,01H,00H,00H,00H,1FH
      C0FB 0000001F

```



## 1-2 パターンの移動

パターンの移動は、いわゆるアニメーションの原理と同じで、連続的にすこしずつ違う形、または、ずらした形を表示していけばよいのです。

現在表示しているパターンを消して、違う形あるいは他の位置にパターンを表示するということを繰り返すわけです。また、パターンを消すには、0をVRAMに書き込めばよいのですが、後に述べる背景との重ね合わせの問題もあるので、一概にはいえません。

### 1-2-1 キー入力

X1には、いくつかキーデータを読む方法がありますが、ここでは、I/Oポートの1900Hを読むという最も簡単な方法を取りました。

```
ld  bc, 1900H
in  a, (c) ; aレジスタに押されたキーの
          ASC11コードが入る
```

なお、X1では、同時に2つ以上のキーを判断できないため（X1 turboは別）、ゲームなどで、ミサイルを打ちながら移動するといったことはできません。

### 1-2-2 基本的移動

リスト1-5は最も基本的なパターンの移動プログラム例です。これは、2, 4, 6, 8キーにより、16×16ドットのパターンを縦横8ドット単位で動かすものです。また、\*キーを押すとシステムに戻ってきます。

移動できるかどうかは、320×200ドットの画面を縦横8



ドット単位で分割して考えているので、縦は 24, 横は 39が  
 最大値であり、更新した X座標, Y座標が画面の範囲を越え  
 ていないかをチェックして判断します。もし、越えていなか  
 れば、現在、表示しているパターンを消去して、新しい位置  
 にパターンを表示します。

リスト I-5

```

1      ;
2      ;      ---      pattern move      ---
3      ;
4  C000      org      0C000H
5      ;
6  C000 010019      key:      ld      bc,1900H      ;bc=i/o port address
7  C003 ED78      in      a,(c)      ;get key (a=ascii code)
8      ;
9  C005 FE32      cp      '2'      ;'2'
10 C007 CA1FC0      jp      z,down      ;if a=2 then down
11 C00A FE34      cp      '4'      ;'4'
12 C00C CA34C0      jp      z,left      ;if a=4 then left
13 C00F FE36      cp      '6'      ;'6'
14 C011 CA48C0      jp      z,right      ;if a=6 then right
15 C014 FE38      cp      '8'      ;'8'
16 C016 CA5DC0      jp      z,up      ;if a=8 then up
17 C019 FE2A      cp      '*'
18 C01B C8      ret      z      ;return to system
19 C01C C300C0      jp      key
20      ;
21 C01F 3A26C1      down:      ld      a,(ypos)      ;y position in a
22 C022 FE17      cp      23      ;y=23 ?
23 C024 D200C0      jp      nc,key      ;if y>23 then key
24 C027 CD88C0      call      erase      ;pattern erase
25 C02A 3C      inc      a      ;y=y+1
26 C02B 3226C1      ld      (ypos),a      ;save y position to a
27 C02E CDCDC0      call      disp      ;pattern display
28 C031 C300C0      jp      key
29      ;
30 C034 3A24C1      left:      ld      a,(xpos)      ;x position in a
31 C037 B7      or      a      ;x=0 ?
32 C038 CA00C0      jp      z,key      ;if x=0 then key
33 C03B CD88C0      call      erase      ;pattern erase
34 C03E 3D      dec      a      ;x=x-1
35 C03F 3224C1      ld      (xpos),a      ;a in (xpos)
36 C042 CDCDC0      call      disp
37 C045 C300C0      jp      key
38      ;
39 C048 3A24C1      right:      ld      a,(xpos)      ;x position in a
40 C04B FE26      cp      38      ;x=38 ?
41 C04D D200C0      jp      nc,key      ;if x>38 then key
42 C050 CD88C0      call      erase      ;pattern erase
43 C053 3C      inc      a      ;x=x+1
44 C054 3224C1      ld      (xpos),a
45 C057 CDCDC0      call      disp
46 C05A C300C0      jp      key
47      ;
48 C05D 3A26C1      up:      ld      a,(ypos)      ;y position in a
49 C060 B7      or      a      ;y=0 ?
50 C061 CA00C0      jp      z,key      ;if y=0 then key
51 C064 CD88C0      call      erase      ;pattern erase
52 C067 3D      dec      a      ;y=y-1
53 C068 3226C1      ld      (ypos),a
54 C06B CDCDC0      call      disp
55 C06E C300C0      jp      key
56      ;
57 C071 3A26C1      adr:      ld      a,(ypos)
58 C074 87      add      a,a
59 C075 6F      ld      l,a
60 C076 2600      ld      h,0
61 C078 1127C1      ld      de,table
    
```



## リスト1-5 つづき

```

62 C07B 19      add    hl,de
63 C07C 4E      ld     c,(hl)
64 C07D 23      inc    hl
65 C07E 46      ld     b,(hl)
66 C07F 2A24C1  ld     hl,(xpos)
67 C082 09      add    hl,bc
68 C083 CBF4    set    6,h
69 C085 44      ld     b,h
70 C086 4D      ld     c,l
71 C087 C9      ret
72              ;
73 C088 F5      erase:  push   af
74 C089 CD71C0  call   adr
75 C08C C5      loop1: push   bc
76 C08D 212800  ld     hl,28H
77 C090 09      add    hl,bc
78 C091 22AEC0  ld     (td+1),hl
79 C094 3E02    ld     a,2
80 C096 F5      loop2: push   af
81 C097 3E08    ld     a,8
82 C099 F5      loop3: push   af
83 C09A AF      xor     a
84 C09B ED79    out    (c),a
85 C09D 03      inc    bc
86 C09E ED79    out    (c),a
87 C0A0 0B      dec    bc
88 C0A1 210008  ld     hl,0800H
89 C0A4 09      add    hl,bc
90 C0A5 44      ld     b,h
91 C0A6 4D      ld     c,l
92 C0A7 F1      pop    af
93 C0A8 3D      dec    a
94 C0A9 C299C0  jp     nz,loop3
95 C0AC F1      pop    af
96 C0AD 010000  td:    ld     bc,0000H
97 C0B0 3D      dec    a
98 C0B1 C296C0  jp     nz,loop2
99 C0B4 C1      pop    bc
100 C0B5 2AAEC0 ld     hl,(td+1)
101 C0B8 7C      ld     a,h
102 C0B9 FEC0    cp     0C0H
103 C0BB D2CBC0  jp     nc,quit1
104 C0BE 210040  ld     hl,4000H
105 C0C1 09      add    hl,bc
106 C0C2 44      ld     b,h
107 C0C3 4D      ld     c,l
108 C0C4 111000  ld     de,10H
109 C0C7 19      add    hl,de
110 C0C8 C38CC0 jp     loop1
111              ;
112 C0CB F1      quit1: pop    af
113 C0CC C9      ret
114              ;
115 C0CD F5      disp:  push   af
116 C0CE CD71C0  call   adr
117 C0D1 2159C1  ld     hl,pdata
118 C0D4 C5      loop4: push   bc
119 C0D5 E5      push   hl
120 C0D6 111000  ld     de,0010H
121 C0D9 19      add    hl,de
122 C0DA EB      ex     de,hl
123 C0DB 212800  ld     hl,0028H
124 C0DE 09      add    hl,bc
125 C0DF 2202C1  ld     (tdl+1),hl
126 C0E2 E1      pop    hl
127              ;
128 C0E3 3E02    ld     a,2
129 C0E5 F5      loop5: push   af
130 C0E6 3E08    ld     a,8
131 C0E8 F5      loop6: push   af
132 C0E9 7E      ld     a,(hl)
133 C0EA ED79    out    (c),a
134 C0EC 03      inc    bc
135 C0ED 1A      ld     a,(de)

```

つづく



136	C0EE	ED79		out	(c),a
137	C0F0	0B		dec	bc
138	C0F1	23		inc	hl
139	C0F2	13		inc	de
140	C0F3	E5		push	hl
141	C0F4	210008		ld	hl,0800H
142	C0F7	09		add	hl,bc
143	C0F8	44		ld	b,h
144	C0F9	4D		ld	c,l
145	C0FA	E1		pop	hl
146	C0FB	F1		pop	af
147	C0FC	3D		dec	a
148	C0FD	C2E8C0		jp	nz,loop6
149	C100	F1		pop	af
150	C101	010000	td1:	ld	bc,0000H
151	C104	3D		dec	a
152	C105	C2E5C0		jp	nz,loop5
153	C108	C1		pop	bc
154	C109	E5		push	hl
155	C10A	2A02C1		ld	hl,(td1+1)
156	C10D	7C		ld	a,h
157	C10E	FEC0		cp	0C0H
158	C110	D221C1		jp	nc,quit2
159	C113	210040		ld	hl,4000H
160	C116	09		add	hl,bc
161	C117	44		ld	b,h
162	C118	4D		ld	c,l
163	C119	E1		pop	hl
164	C11A	111000		ld	de,10H
165	C11D	19		add	hl,de
166	C11E	C3D4C0		jp	loop4
167					
168	C121	E1	quit2:	pop	hl
169	C122	F1		pop	af
170	C123	C9		ret	
171					
172	C124	0000	xpos:	defw	0
173	C126	00	ypos:	defb	0
174					
175	C127	00002800	table:	defw	0,40,80,120,160,200,240,280,320,360
	C12B	50007800			
	C12F	A000C800			
	C133	F0001801			
	C137	40016801			
176	C13B	9001B801		defw	400,440,480,520,560,600,640,680,720
	C13F	E0010802			
	C143	30025802			
	C147	8002A802			
	C14B	D002			
177	C14D	F8022003		defw	760,800,840,880,920,960
	C151	48037003			
	C155	9803C003			
178					
179	C159	000A3F3F	pdata:	defb	00H,0AH,3FH,3FH,3FH,3DH,3DH,3FH
	C15D	3F3D3D3F			
180	C161	3F3C3C3F		defb	3FH,3CH,3CH,3FH,3FH,3FH,3FH,00H
	C165	3F3F3F00			
181	C169	00A0FCFC		defb	00H,0A0H,0FCH,0FCH,0FCH,0BCH,0BCH,0FCH
	C16D	FCBCBCFC			
182	C171	FC3C3CFC		defb	0FCH,3CH,3CH,0FCH,0FCH,0FCH,0FCH,00H
	C175	FCFCFC00			
183					
184	C179	073066CE		defb	07H,30H,66H,0CEH,0CEH,4CH,0CH,0EH
	C17D	CE4C0C0E			
185	C181	46DBDB80		defb	46H,0DBH,0DBH,80H,00H,00H,00H,0F9H
	C185	000000F9			
186	C189	E00C6673		defb	0E0H,0CH,66H,73H,73H,32H,30H,70H
	C18D	73323070			
187	C191	62DBDB01		defb	62H,0DBH,0DBH,01H,00H,00H,00H,9FH
	C195	0000009F			
188					
189	C199	073F66CE		defb	07H,3FH,66H,0CEH,0CEH,4CH,0CH,0EH
	C19D	CE4C0C0E			



リスト1-5 つづき

190	C1A1	46C3C380	defb	46H,0C3H,0C3H,80H,00H,00H,00H,0F8H
	C1A5	000000F8		
191	C1A9	E0FC6673	defb	0E0H,0FCH,66H,73H,73H,32H,30H,70H
	C1AD	73323070		
192	C1B1	62C3C301	defb	62H,0C3H,0C3H,01H,00H,00H,00H,1FH
	C1B5	0000001F		

## 1-2-3 大きなパターンの移動

これまで扱ってきた16×16ドットのパターンは、ゲームに出てくる普通のキャラクタとしては、十分な大きさですが、やはり、見た目の面白さを出すためにはより大きなパターンを表示したいところです。そこで、こんどは64×64ドットの大きなパターンを動かしてみます。

パターンがこれだけ大きいとデータ量も相当なものです。ですから、大きなパターンを表示、移動させる場合、いちいち消去と表示の繰り返しをしていたのでは、大幅に遅くなってしまいます。そのため、パターンがちらつくこともあります。この対策として『部分消去』という方法を使います。

これは、たとえば左へ8ドット移動するときはパターンの右8ドットだけを消去し、あとは、移動する位置へパターンのデータを書き込むだけにするという方法です。

リスト1-6の、‘erase 1’と‘erase 2’は、それぞれ左右8ドットを消去するもの、上下の8ラインを消去するものです。

このような大きなパターンでなくてもちらつきがでる場合があります。これは、前に表示してるものと同じものを同じところに書き込む場合に起こります。これを解決するのは、アルゴリズム上の問題で、キャラクタが動かないときは再表示しないというようにすればよいでしょう。



リスト1-6

```

1      ;
2      ;      pattern print (64*64 dot)
3      ;
4  C000      org      0C000H
5      ;
6  C000 2103C2      ld      hl,pdata
7  C003 1104C2      ld      de,pdata+1
8  C006 010006      ld      bc,1536
9  C009 36FF        ld      (hl),0FFH
10 C00B EDB0        ldir
11      ;
12 C00D 010019      key:   ld      bc,1900H
13 C010 ED78        in      a,(c)
14      ;
15 C012 FE32        cp      '2'
16 C014 CA2CC0      jp      z,down
17 C017 FE34        cp      '4'
18 C019 CA4FC0      jp      z,left
19 C01C FE36        cp      '6'
20 C01E CA6FC0      jp      z,right
21 C021 FE38        cp      '8'
22 C023 CA8CC0      jp      z,up
23 C026 FE2A        cp      '*'
24 C028 C8          ret     z
25 C029 C30DC0      jp      key
26      ;
27 C02C 3AFFC1      down:  ld      a,(ypos)
28 C02F FE11        cp      17
29 C031 D20DC0      jp      nc,key
30 C034 3C          inc     a
31 C035 32FFC1      ld      (ypos),a
32 C038 2103C2      ld      hl,pdata
33 C03B CDC8C0      call   displ
34 C03E 2A01C2      ld      hl,(adwork)
35 C041 012800      ld      bc,28H
36 C044 AF          xor     a           ;carry flag clear
37 C045 ED42        sbc     hl,bc
38 C047 44          ld      b,h
39 C048 4D          ld      c,l
40 C049 CD80C1      call   erase2
41 C04C C30DC0      jp      key
42      ;
43 C04F 3AFDC1      left:  ld      a,(xpos)
44 C052 B7          or      a
45 C053 CA0DC0      jp      z,key
46 C056 3D          dec     a
47 C057 32FDC1      ld      (xpos),a
48 C05A 2103C2      ld      hl,pdata
49 C05D CDC8C0      call   displ
50 C060 2A01C2      ld      hl,(adwork)
51 C063 110800      ld      de,8
52 C066 19          add     hl,de
53 C067 44          ld      b,h
54 C068 4D          ld      c,l
55 C069 CD2EC1      call   erase1
56 C06C C30DC0      jp      key
57      ;
58 C06F 3AFDC1      right: ld      a,(xpos)
59 C072 FE20        cp      32
60 C074 D20DC0      jp      nc,key
61 C077 3C          inc     a
62 C078 32FDC1      ld      (xpos),a
63 C07B 2103C2      ld      hl,pdata
64 C07E CDC8C0      call   displ
65 C081 ED4B01C2    ld      bc,(adwork)

```



## リストI-6 つづき

```

66 C085 0B          dec    bc
67 C086 CD2EC1      call   erase1
68 C089 C30DC0      jp      key
69                  ;
70 C08C 3AFFC1      up:    ld     a,(ypos)
71 C08F B7          or     a
72 C090 CA0DC0      jp      z,key
73 C093 3D          dec    a
74 C094 32FFC1      ld     (ypos),a
75 C097 2103C2      ld     hl,pdata
76 C09A CDC8C0      call   disp1
77 C09D 2A01C2      ld     hl,(adwork)
78 C0A0 114001      ld     de,320
79 C0A3 19          add    hl,de
80 C0A4 44          ld     b,h
81 C0A5 4D          ld     c,l
82 C0A6 CD80C1      call   erase2
83 C0A9 C30DC0      jp      key
84                  ;
85                  ;
86 C0AC E5          adr:   push   hl
87 C0AD 01CBC1      ld     bc,ydata
88 C0B0 2AFFC1      ld     hl,(ypos)      ;y position /8
89 C0B3 29          add    hl,hl
90 C0B4 09          add    hl,bc
91 C0B5 5E          ld     e,(hl)
92 C0B6 23          inc    hl
93 C0B7 56          ld     d,(hl)
94 C0B8 2AFDC1      ld     hl,(xpos)      ;x position /8
95 C0BB 19          add    hl,de      ;address
96 C0BC 110040      ld     de,4000H      ;blue vram base address
97 C0BF 19          add    hl,de      ;blue address
98 C0C0 E5          push   hl
99 C0C1 C1          pop    bc
100 C0C2 E1         pop    hl
101 C0C3 ED4301C2   ld     (adwork),bc
102 C0C7 C9         ret
103                ;
104 C0C8 CDACC0      disp1: call   adr
105 C0CB 3E08        loop1: ld     a,8
106 C0CD C5          push   bc
107 C0CE C5          loop2: push  bc
108 C0CF CD1DC1      call   disp2
109 C0D2 CBD8        set    3,b
110 C0D4 CD1DC1      call   disp2
111 C0D7 CBE0        set    4,b
112 C0D9 CB98        res    3,b
113 C0DB CD1DC1      call   disp2
114 C0DE CBD8        set    3,b
115 C0E0 CD1DC1      call   disp2
116 C0E3 CBE8        set    5,b
117 C0E5 CBA0        res    4,b
118 C0E7 CB98        res    3,b
119 C0E9 CD1DC1      call   disp2
120 C0EC CBD8        set    3,b
121 C0EE CD1DC1      call   disp2
122 C0F1 CBE0        set    4,b
123 C0F3 CB98        res    3,b
124 C0F5 CD1DC1      call   disp2
125 C0F8 CBD8        set    3,b
126 C0FA CD1DC1      call   disp2
127 C0FD D1         pop    de
128 C0FE E5          push   hl
129 C0FF 212800      ld     hl,28H
130 C102 19          add    hl,de
131 C103 44          ld     b,h
132 C104 4D          ld     c,l

```

つづく



リストI-6 つづき

133	C105	E1		pop	hl
134	C106	3D		dec	a
135	C107	20C5		jr	nz,loop2
136			:		
137	C109	C1		pop	bc
138	C10A	E5		push	hl
139	C10B	78		ld	a,b
140	C10C	FEC0		cp	0C0H
141	C10E	D21BC1		jp	nc,ret
142	C111	210040		ld	hl,4000H
143	C114	09		add	hl,bc
144	C115	44		ld	b,h
145	C116	4D		ld	c,l
146	C117	E1		pop	hl
147	C118	C3CBC0		jp	loop1
148	C11B	E1	ret:	pop	hl
149	C11C	C9		ret	
150			:		
151	C11D	F5	disp2:	push	af
152	C11E	3E08		ld	a.8
153	C120	C5		push	bc
154	C121	F5	loop3:	push	af
155	C122	7E		ld	a,(hl)
156	C123	ED79		out	(c),a
157	C125	03		inc	bc
158	C126	23		inc	hl
159	C127	F1		pop	af
160	C128	3D		dec	a
161	C129	20F6		jr	nz,loop3
162	C12B	C1		pop	bc
163	C12C	F1		pop	af
164	C12D	C9		ret	
165			:		
166	C12E	3E08	erasel:	ld	a.8
167	C130	C5		push	bc
168	C131	C5	loop5:	push	bc
169	C132	CD7AC1		call	disp3
170	C135	CBD8		set	3,b
171	C137	CD7AC1		call	disp3
172	C13A	CBE0		set	4,b
173	C13C	CB98		res	3,b
174	C13E	CD7AC1		call	disp3
175	C141	CBD8		set	3,b
176	C143	CD7AC1		call	disp3
177	C146	CBE8		set	5,b
178	C148	CBA0		res	4,b
179	C14A	CB98		res	3,b
180	C14C	CD7AC1		call	disp3
181	C14F	CBD8		set	3,b
182	C151	CD7AC1		call	disp3
183	C154	CBE0		set	4,b
184	C156	CB98		res	3,b
185	C158	CD7AC1		call	disp3
186	C15B	CBD8		set	3,b
187	C15D	CD7AC1		call	disp3
188	C160	D1		pop	de
189	C161	E5		push	hl
190	C162	212800		ld	hl,28H
191	C165	19		add	hl,de
192	C166	44		ld	b,h
193	C167	4D		ld	c,l
194	C168	E1		pop	hl
195	C169	3D		dec	a
196	C16A	20C5		jr	nz,loop5
197			:		
198	C16C	C1		pop	bc
199	C16D	78		ld	a,b

つづく



## リストI-6 つづき

```

200 C16E FEC0      cp      0C0H
201 C170 D0        ret     nc
202 C171 210040    ld      hl,4000H
203 C174 09        add     hl,bc
204 C175 44        ld      b,h
205 C176 4D        ld      c,l
206 C177 C32EC1    jp      erase1
207                ;
208 C17A F5        disp3:  push  af
209 C17B AF        xor     a
210 C17C ED79      out     (c),a
211 C17E F1        pop     af
212 C17F C9        ret
213                ;
214 C180 C5        erase2:  push  bc
215 C181 CDBDC1    call    disp4
216 C184 CBD8      set     3,b
217 C186 CDBDC1    call    disp4
218 C189 CBE0      set     4,b
219 C18B CB98      res     3,b
220 C18D CDBDC1    call    disp4
221 C190 CBD8      set     3,b
222 C192 CDBDC1    call    disp4
223 C195 CBE8      set     5,b
224 C197 CBA0      res     4,b
225 C199 CB98      res     3,b
226 C19B CDBDC1    call    disp4
227 C19E CBD8      set     3,b
228 C1A0 CDBDC1    call    disp4
229 C1A3 CBE0      set     4,b
230 C1A5 CB98      res     3,b
231 C1A7 CDBDC1    call    disp4
232 C1AA CBD8      set     3,b
233 C1AC CDBDC1    call    disp4
234                ;
235 C1AF C1        pop     bc
236 C1B0 78        ld      a,b
237 C1B1 FEC0      cp      0C0H
238 C1B3 D0        ret     nc
239 C1B4 210040    ld      hl,4000H
240 C1B7 09        add     hl,bc
241 C1B8 44        ld      b,h
242 C1B9 4D        ld      c,l
243 C1BA C380C1    jp      erase2
244                ;
245 C1BD 3E08      disp4:  ld      a,8
246 C1BF C5        push    bc
247 C1C0 F5        loop7:  push  af
248 C1C1 AF        xor     a
249 C1C2 ED79      out     (c),a
250 C1C4 03        inc     bc
251 C1C5 F1        pop     af
252 C1C6 3D        dec     a
253 C1C7 20F7      jr      nz,loop7
254 C1C9 C1        pop     bc
255 C1CA C9        ret
256                ;
257 C1CB 00002800   ydata:  defw    0,40,80,120,160,200,240,280,320,360
      C1CF 50007800
      C1D3 A000C800
      C1D7 F0001801
      C1DB 40016801
258 C1DF 9001B801   defw    400,440,480,520,560,600,640,680,720
      C1E3 E0010802
      C1E7 30025802
      C1EB 8002A802
      C1EF D002

```

つづく



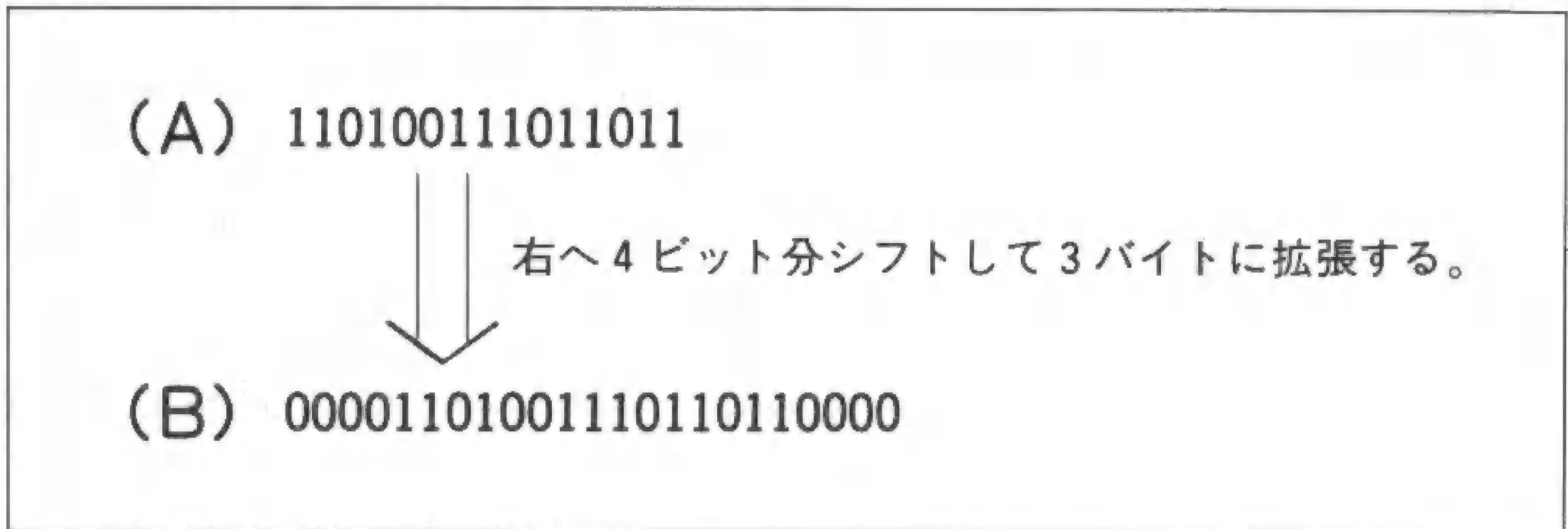
```
259 C1F1 F8022003          defw    760,800,840,880,920,960
    C1F5 48037003
    C1F9 9803C003
260                               ;
261 C1FD 0800          xpos:    defw    8
262 C1FF 0000          ypos:    defw    0
263 C201 0000          adwork:  defw    0
264                               ;
265 C203                pdata:
```

## 1-2-4 横4ビット単位の移動

VRAMの構造上、上下には1ドット単位でも移動できますが、左右へは8ドット単位の方が簡単です。しかし、8ドット単位というのは、場合によっては移動量が大きすぎて、なめらかに移動しているようには見えません。そこで、ここでは横4ドット単位に移動する例をみていきます。

いま、横16ドット（2バイト）のパターンがあるとして、このパターン・データを rld 命令か rrd 命令によって、4ビット分シフトして3バイトのデータにします（図1-4）。そして、図の（A）のデータをVRAMに書き込んだ後、右へ4ビット移動するには（B）のデータを同じVRAMのアドレスに書き込めばよいわけです。さらに、右へ4ビット移動するには、再び、（A）のデータを前のVRAMのアドレス+1に書き込みます。例をリスト1-7に示します。

図1-4 2バイトから3バイトへ拡張





## リスト1-7

```

1      :
2      :---- pattern move (4 dot) ----
3      :
4  C000      org      0C000H
5      :
6  C000 CDB9C1      call      shift
7      :
8  C003 010019      key:      ld      bc,1900H
9  C006 ED78          in      a,(c)
10     :
11  C008 FE32          cp      '2'
12  C00A CA22C0        jp      z,down
13  C00D FE34          cp      '4'
14  C00F CA39C0        jp      z,left
15  C012 FE36          cp      '6'
16  C014 CA68C0        jp      z,right
17  C017 FE38          cp      '8'
18  C019 CA97C0        jp      z,up
19  C01C FE2A          cp      '*'
20  C01E C8            ret      z
21  C01F C303C0        jp      key
22     :
23  C022 3A3BC1      down:      ld      a,(ypos)
24  C025 FE17          cp      23
25  C027 D203C0        jp      nc,key
26  C02A F5            push     af
27  C02B CDC3C0        call     erase
28  C02E F1            pop      af
29  C02F 3C            inc      a
30  C030 323BC1        ld      (ypos),a
31  C033 CD3DC1        call     disp
32  C036 C303C0        jp      key
33     :
34  C039 3A39C1      left:      ld      a,(xpos)
35  C03C B7            or      a
36  C03D 2007          jr      nz,left1      ;If xpos>0 then left1
37  C03F 3A37C1        ld      a,(flg)      ;flg in a
38  C042 B7            or      a            ;flg=0 ?
39  C043 CA03C0        jp      z,key        ;If flg=0 then key
40  C046 CDC3C0      left1:      call     erase      ;Erase pattern
41  C049 3A37C1        ld      a,(flg)
42  C04C B7            or      a
43  C04D C25EC0        jp      nz,left2      ;If flg=0ffh then left2
44  C050 3A39C1        ld      a,(xpos)
45  C053 3D            dec      a            ;decrement xpos
46  C054 3239C1        ld      (xpos),a
47  C057 3EFF          ld      a,0FFH        ;set flg
48  C059 3237C1        ld      (flg),a
49  C05C 1804          jr      left3
50  C05E AF            left2:      xor      a            ;clear flg
51  C05F 3237C1        ld      (flg),a
52  C062 CD3DC1      left3:      call     disp
53  C065 C303C0        jp      key
54     :
55  C068 3A39C1      right:      ld      a,(xpos)
56  C06B FE26          cp      38            ;xpos=38 ?
57  C06D 3807          jr      c,right1      ;If xpos>38 then right1
58  C06F 3A37C1        ld      a,(flg)
59  C072 B7            or      a
60  C073 CA03C0        jp      z,key
61  C076 CDC3C0      right1:      call     erase
62  C079 3A37C1        ld      a,(flg)
63  C07C B7            or      a
64  C07D 280D          jr      z,right2      ;If flg=0 then right2
65  C07F AF            xor      a            ;Clear flg
66  C080 3237C1        ld      (flg),a
67  C083 3A39C1        ld      a,(xpos)
68  C086 3C            inc      a            ;Increment flg
69  C087 3239C1        ld      (xpos),a
70  C08A 1805          jr      right3
71  C08C 3EFF          right2:      ld      a,0FFH        ;Set flg
72  C08E 3237C1        ld      (flg),a
73  C091 CD3DC1      right3:      call     disp
74  C094 C303C0        jp      key

```

つづく



リスト1-7

つづき

```

75      ;
76 C097 3A3BC1 up: ld a,(ypos)
77 C09A B7      or a
78 C09B CA03C0  jp z,key
79 C09E F5      push af
80 C09F CDC3C0  call erase
81 C0A2 F1      pop af
82 C0A3 3D      dec a
83 C0A4 323BC1  ld (ypos),a
84 C0A7 CD3DC1  call disp
85 C0AA C303C0  jp key
86      ;
87 C0AD 01F3C1  adr: ld bc,table
88 C0B0 2A3BC1  ld hl,(ypos)
89 C0B3 29      add hl,hl
90 C0B4 09      add hl,bc
91 C0B5 5E      ld e,(hl)
92 C0B6 23      inc hl
93 C0B7 56      ld d,(hl)
94 C0B8 2A39C1  ld hl,(xpos)
95 C0BB 19      add hl,de
96 C0BC 110040  ld de,4000H
97 C0BF 19      add hl,de
98 C0C0 E5      push hl
99 C0C1 C1      pop bc
100 C0C2 C9     ret
101      ;
102      ; ERASE PATTERN
103      ;
104 C0C3 CD15C3  erase: call dely          ;Delay
105 C0C6 3A37C1  ld a,(flg)
106 C0C9 B7      or a
107 C0CA 2804      jr z,erase1
108 C0CC 3E03      ld a,3          ;Pattern length (3byte) in a
109 C0CE 1802      jr erase2
110 C0D0 3E02      erase1: ld a,2          ;Pattern length (2byte) in a
111 C0D2 3238C1  erase2: ld (leng),a      ;Erase pattern
112 C0D5 CDADC0  call adr
113 C0D8 ED4333C1 ld (adwork),bc          ;Address of vram in adwork
114 C0DC CDFEC0  call erplan          ;Blue plane
115      ;
116 C0DF ED4B33C1 ld bc,(adwork)
117 C0E3 3E40      ld a,40H
118 C0E5 80      add a,b          ;Red plane
119 C0E6 47      ld b,a
120 C0E7 ED4333C1 ld (adwork),bc
121 C0EB CDFEC0  call erplan
122      ;
123 C0EE ED4B33C1 ld bc,(adwork)
124 C0F2 3E40      ld a,40H
125 C0F4 80      add a,b          ;Green plane
126 C0F5 47      ld b,a
127 C0F6 ED4333C1 ld (adwork),bc
128 C0FA CDFEC0  call erplan
129 C0FD C9      ret
130      ;
131 C0FE CD0FC1  erplan: call er8lin
132 C101 ED4B33C1 ld bc,(adwork)
133 C105 212800  ld hl,40
134 C108 09      add hl,bc
135 C109 44      ld b,h
136 C10A 4D      ld c,l
137 C10B CD12C1  call @er8li
138 C10E C9      ret
139      ;
140 C10F 2A33C1  er8lin: ld hl,(adwork) ;Erase upper 8line
141 C112 2235C1  @er8li: ld (adw1),hl   ;Erase lower 8line
142 C115 1608      ld d,8          ;Set counter
143 C117 3A38C1  er8li: ld a,(leng)
144 C11A 5F      ld e,a
145 C11B AF      xor a          ;clear a
146 C11C ED79      er8li2: out (c),a
147 C11E 03      inc bc
148 C11F 1D      dec e          ;decrement length
149 C120 20FA      jr nz,er8li2
150      ;

```

つづく



## リスト1-7 つづき

```

151 C122 15          dec    d          ;decrement counter
152 C123 C8          ret     z
153 C124 ED4B35C1    ld     bc,(adw1)
154 C128 210008      ld     hl,0800H
155 C12B 09          add     hl,bc
156 C12C 2235C1      ld     (adw1),hl
157 C12F 44          ld     b,h
158 C130 4D          ld     c,l
159 C131 18E4        jr     er811
160
161 C133              ;
162 C135              adwork: defs    2
163 C137 00          adw1:  defs    2
164 C138 00          flg:   defb    0
165 C139 0A00        leng:  defb    0
166 C13B 0000        xpos:  defw    10
167                  ypos:  defw    0
168                  ;
169                  ; display
170 C13D CD15C3      disp:  call    dely
171 C140 3A37C1      ld     a,(flg)
172 C143 B7          or     a
173 C144 2808        jr     z,disp1
174 C146 3E03        ld     a,3          ;Pattern length in a
175 C148 DD2185C2    ld     ix,sdata      ;Address of sdata in ix
176 C14C 1806        jr     disp2
177 C14E 3E02        disp1: ld     a,2      ;Pattern length in a
178 C150 DD2125C2    ld     ix,pdata      ;Address of pdata in ix
179 C154 3238C1      disp2: ld     (leng),a
180 C157 CDADC0      call    adr
181 C15A ED4333C1    ld     (adwork),bc
182 C15E CD80C1      call    dpplan      ;Blue plane
183
184 C161 ED4B33C1    ld     bc,(adwork)
185 C165 3E40        ld     a,40H
186 C167 80          add     a,b          ;Red plane
187 C168 47          ld     b,a
188 C169 ED4333C1    ld     (adwork),bc
189 C16D CD80C1      call    dpplan
190
191 C170 ED4B33C1    ld     bc,(adwork)
192 C174 3E40        ld     a,40H
193 C176 80          add     a,b          ;Green plane
194 C177 47          ld     b,a
195 C178 ED4333C1    ld     (adwork),bc
196 C17C CD80C1      call    dpplan
197 C17F C9          ret
198
199 C180 CD91C1      dpplan: call    dp81in
200 C183 ED4B33C1    ld     bc,(adwork)
201 C187 212800      ld     hl,40
202 C18A 09          add     hl,bc
203 C18B 44          ld     b,h
204 C18C 4D          ld     c,l
205 C18D CD94C1      call    @dp81i
206 C190 C9          ret
207
208 C191 2A33C1      dp81in: ld     hl,(adwork)
209 C194 2235C1      @dp81i: ld     (adw1),hl
210 C197 1608        ld     d,8
211 C199 3A38C1      dp811:  ld     a,(leng)
212 C19C 5F          ld     e,a
213 C19D DD7E00      dp812:  ld     a,(ix)
214 C1A0 ED79        out     (c),a
215 C1A2 03          inc     bc
216 C1A3 DD23        inc     ix
217 C1A5 1D          dec     e
218 C1A6 20F5        jr     nz,dp812
219
220 C1A8 15          dec     d
221 C1A9 C8          ret     z
222 C1AA ED4B35C1    ld     bc,(adw1)
223 C1AE 210008      ld     hl,0800H
224 C1B1 09          add     hl,bc
225 C1B2 2235C1      ld     (adw1),hl
226 C1B5 44          ld     b,h

```

つづく



リストI-7 つづき

```

227 C1B6 4D          ld      c,l
228 C1B7 18E0        jr      dp811
229                  ;
230 C1B9 DD2125C2    shift: ld      ix,pdata
231 C1BD FD2185C2    ld      iy,sdata
232 C1C1 21F1C1      ld      hl,wdata
233 C1C4 0630        ld      b,48
234 C1C6 DD5E00      jmp1:  ld      e,(ix)
235 C1C9 DD5601      ld      d,(ix+1)
236 C1CC ED53F1C1    ld      (wdata),de
237 C1D0 AF          xor      a
238 C1D1 ED6F        rld
239 C1D3 FD7700      ld      (iy),a
240 C1D6 FD23        inc      iy
241 C1D8 7E          ld      a,(hl)
242 C1D9 23          inc      hl
243 C1DA ED6F        rld
244 C1DC FD7700      ld      (iy),a
245 C1DF FD23        inc      iy
246 C1E1 7E          ld      a,(hl)
247 C1E2 E6F0        and      0F0H
248 C1E4 FD7700      ld      (iy),a
249 C1E7 FD23        inc      iy
250 C1E9 DD23        inc      ix
251 C1EB DD23        inc      ix
252 C1ED 2B          dec      hl
253 C1EE 10D6        djnz    jmp1
254 C1F0 C9          ret
255                  ;
256 C1F1              wdata: defs    2
257                  ;
258 C1F3 00002800    table: defw    0,40,80,120,160,200,240,280,320,360,400
C1F7 50007800
C1FB A000C800
C1FF F0001801
C203 40016801
C207 9001
259 C209 B801E001    defw    440,480,520,560,600,640,680,720,760,800
C20D 08023002
C211 58028002
C215 A802D002
C219 F8022003
260 C21D 48037003    defw    840,880,920,960
C221 9803C003
261                  ;
262 C225 00000AA0    pdata: defb    00H,00H,0AH,0A0H,3FH,0FCH,3FH,0FCH    ;blue
C229 3FFC3FFC
263 C22D 3FFC3DBC    defb    3FH,0FCH,3DH,0BCH,3DH,0BCH,3FH,0FCH
C231 3DBC3FFC
264 C235 3FFC3C3C    defb    3FH,0FCH,3CH,3CH,3CH,3CH,3FH,0FCH
C239 3C3C3FFC
265 C23D 3FFC3FFC    defb    3FH,0FCH,3FH,0FCH,3CH,0FCH,00H,00H
C241 3CFC0000
266                  ;
267 C245 07E0300C    defb    07H,0E0H,30H,0CH,66H,66H,0CEH,73H    ;red
C249 6666CE73
268 C24D CE734C32    defb    0CEH,73H,4CH,32H,0CH,30H,0EH,70H
C251 0C300E70
269 C255 4662DBDB    defb    46H,62H,0DBH,0DBH,0DBH,0DBH,80H,01H
C259 DBDB8001
270 C25D 00000000    defb    00H,00H,00H,00H,00H,00H,0F9H,9FH
C261 0000F99F
271                  ;
272 C265 07E03FFC    defb    07H,0E0H,3FH,0FCH,66H,66H,0CEH,73H    ;green
C269 6666CE73
273 C26D CE734C32    defb    0CEH,73H,4CH,32H,0CH,30H,0EH,70H
C271 0C300E70
274 C275 4662C3C3    defb    46H,62H,0C3H,0C3H,0C3H,0C3H,80H,01H
C279 C3C38001
275 C27D 00000000    defb    00H,00H,00H,00H,00H,00H,0F8H,1FH
C281 0000F81F
276                  ;
277 C285              sdata: defs    144
278                  ;
279 C315 010010      dely:  ld      bc,1000H
280 C318 0B          dely1: dec      bc

```

つづく



## リスト1-7 つづき

281	C319 79	ld	a,c
282	C31A B0	or	b
283	C31B C8	ret	z
284	C31C 18FA	jr	delyl



## 1-3 応用テクニック

これまでは、パターンの表示や消去のときに背景のことを考慮していませんでした。そのため、背景に何か表示されている場合、これまでの方法では、背景に影響を与えてしまいます。ここではこの背景との重ね合わせテクニックと PCG によるスクロールを紹介します。

### 1-3-1 重ね合わせ

テキスト画面とグラフィック画面のそれぞれに移動物と背景を表示するぶんには、重ね合わせを気にする必要はありません。しかし、グラフィック画面だけを使うときは、背景に影響を与えないようにする必要があります。これには、パターンを消すときに背景を書き直せばよいのですが、こうするとパターンのデータの‘0’の部分だけ背景が黒くなってしまい、多少不自然な感じがします。これを避けるためには次のような方法があります。

〔1〕パターンと背景の排他的論理和をとる

Z80のマシン語に‘xor’というニモニックがありますが、これは同じ値どうして演算すると0になります。これを利用すると背景に影響を与えずパターンを表示することができます。

まず、表示するときは、その位置の背景データとパターン・データのXORをとってVRAMに書き込みます。消すときは、さきほど論理演算した結果とパターン・データのXORをさらにとって、そのデータを書き込みます（背景を書き直してもよい）。

リスト1-8にこの例を示します。実行前にグラフィックで適当に背景を書いておいてください。実行するとわかりますが、このプログラムでは背景に影響を与えないかわりに、パ



ターンそのものの色が背景の色によって変化してしまうという欠点を持っています。

## リストI-8

```

1      :
2      :      ---      pattern move (xor)      ---
3      :
4  C000      org      0C000H
5      :
6  C000 CD8BC0      call      erase
7      :
8  C003 010019      key:      ld      bc,1900H
9  C006 ED78          in      a,(c)
10     :
11  C008 FE32          cp      '2'
12  C00A CA22C0        jp      z,down
13  C00D FE34          cp      '4'
14  C00F CA37C0        jp      z,left
15  C012 FE36          cp      '6'
16  C014 CA4BC0        jp      z,right
17  C017 FE38          cp      '8'
18  C019 CA60C0        jp      z,up
19  C01C FE2A          cp      '*'
20  C01E C8            ret      z
21  C01F C303C0        jp      key
22     ;
23  C022 3A47C1        down:      ld      a,(ypos)
24  C025 FE17          cp      23
25  C027 D203C0        jp      nc,key
26  C02A CD8BC0        call      erase
27  C02D 3C            inc      a
28  C02E 3247C1        ld      (ypos),a
29  C031 CDE8C0        call      disp
30  C034 C303C0        jp      key
31     ;
32  C037 3A45C1        left:      ld      a,(xpos)
33  C03A B7            or      a
34  C03B CA03C0        jp      z,key
35  C03E CD8BC0        call      erase
36  C041 3D            dec      a
37  C042 3245C1        ld      (xpos),a
38  C045 CDE8C0        call      disp
39  C048 C303C0        jp      key
40     ;
41  C04B 3A45C1        right:     ld      a,(xpos)
42  C04E FE26          cp      38
43  C050 D203C0        jp      nc,key
44  C053 CD8BC0        call      erase
45  C056 3C            inc      a
46  C057 3245C1        ld      (xpos),a
47  C05A CDE8C0        call      disp
48  C05D C303C0        jp      key
49     ;
50  C060 3A47C1        up:        ld      a,(ypos)
51  C063 B7            or      a
52  C064 CA03C0        jp      z,key
53  C067 CD8BC0        call      erase
54  C06A 3D            dec      a
55  C06B 3247C1        ld      (ypos),a
56  C06E CDE8C0        call      disp
57  C071 C303C0        jp      key
58     ;
59  C074 3A47C1        adr:        ld      a,(ypos)
60  C077 87            add      a,a
61  C078 6F            ld      l,a
62  C079 2600          ld      h,0
63  C07B 1148C1        ld      de,table
64  C07E 19            add      hl,de
65  C07F 4E            ld      c,(hl)
66  C080 23            inc      hl
67  C081 46            ld      b,(hl)

```



リストI-8

つづき

68	C082	2A45C1	ld	hl,(xpos)	
69	C085	09	add	hl,bc	
70	C086	CBF4	set	6,h	
71	C088	44	ld	b,h	
72	C089	4D	ld	c,l	
73	C08A	C9	ret		
74			:		
75	C08B	F5	erase: push	af	
76	C08C	CD74C0	call	adr	
77	C08F	217AC1	ld	hl,pdata	
78	C092	C5	loop1: push	bc	
79	C093	E5	push	hl	
80	C094	111000	ld	de,0010H	
81	C097	19	add	hl,de	
82	C098	EB	ex	de,hl	
83	C099	212800	ld	hl,28H	
84	C09C	09	add	hl,bc	
85	C09D	22C6C0	ld	(td+1),hl	
86	C0A0	E1	pop	hl	
87	C0A1	3E02	ld	a,2	
88	C0A3	F5	loop2: push	af	
89	C0A4	3E08	ld	a,8	
90	C0A6	F5	loop3: push	af	
91	C0A7	ED78	in	a,(c)	:read buck ground
92	C0A9	AE	xor	(hl)	:((buck ground) xor (pattern deta)
93	C0AA	ED79	out	(c),a	:print
94	C0AC	03	inc	bc	:increment address of vram
95	C0AD	ED78	in	a,(c)	
96	C0AF	EB	ex	de,hl	
97	C0B0	AE	xor	(hl)	
98	C0B1	ED79	out	(c),a	
99	C0B3	EB	ex	de,hl	
100	C0B4	0B	dec	bc	
101	C0B5	23	inc	hl	
102	C0B6	13	inc	de	
103	C0B7	E5	push	hl	
104	C0B8	210008	ld	hl,0800H	
105	C0BB	09	add	hl,bc	
106	C0BC	44	ld	b,h	
107	C0BD	4D	ld	c,l	
108	C0BE	E1	pop	hl	
109	C0BF	F1	pop	af	
110	C0C0	3D	dec	a	
111	C0C1	C2A6C0	jp	nz,loop3	
112	C0C4	F1	pop	af	
113	C0C5	010000	td: ld	bc,0000H	
114	C0C8	3D	dec	a	
115	C0C9	C2A3C0	jp	nz,loop2	
116	C0CC	C1	pop	bc	
117	C0CD	E5	push	hl	
118	C0CE	2AC6C0	ld	hl,(td+1)	
119	C0D1	7C	ld	a,h	
120	C0D2	FEC0	cp	0C0H	
121	C0D4	D2E5C0	jp	nc,quit1	
122	C0D7	210040	ld	hl,4000H	
123	C0DA	09	add	hl,bc	
124	C0DB	E5	push	hl	
125	C0DC	C1	pop	bc	
126	C0DD	E1	pop	hl	
127	C0DE	111000	ld	de,10H	
128	C0E1	19	add	hl,de	
129	C0E2	C392C0	jp	loop1	
130			:		
131	C0E5	E1	quit1: pop	hl	
132	C0E6	F1	pop	af	
133	C0E7	C9	ret		
134			:		
135	C0E8	F5	disp: push	af	
136	C0E9	CD74C0	call	adr	
137	C0EC	217AC1	ld	hl,pdata	
138	C0EF	C5	loop4: push	bc	
139	C0F0	E5	push	hl	
140	C0F1	111000	ld	de,0010H	
141	C0F4	19	add	hl,de	
142	C0F5	EB	ex	de,hl	
143	C0F6	212800	ld	hl,0028H	

つづく



## リストI-8 つづき

```

144 C0F9 09          add    hl,bc
145 C0FA 2223C1      ld      (td1+1),hl
146 C0FD E1          pop     hl
147                  ;
148 C0FE 3E02          ld      a,2
149 C100 F5          loop5:  push   af
150 C101 3E08          ld      a,8
151 C103 F5          loop6:  push   af
152 C104 ED78          in      a,(c)          ;Read buck ground data
153 C106 AE          xor      (hl)          ;xor (buck) and (pattern data)
154 C107 ED79          out     (c),a
155 C109 03          inc     bc
156 C10A ED78          in      a,(c)
157 C10C EB          ex      de,hl
158 C10D AE          xor      (hl)
159 C10E ED79          out     (c),a
160 C110 EB          ex      de,hl
161 C111 0B          dec     bc
162 C112 23          inc     hl
163 C113 13          inc     de
164 C114 E5          push    hl
165 C115 210008      ld      hl,0800H
166 C118 09          add     hl,bc
167 C119 44          ld      b,h
168 C11A 4D          ld      c,l
169 C11B E1          pop     hl
170 C11C F1          pop     af
171 C11D 3D          dec     a
172 C11E C203C1      jp      nz,loop6
173 C121 F1          pop     af
174 C122 010000      tdl:    ld      bc,0000H
175 C125 3D          dec     a
176 C126 C200C1      jp      nz,loop5
177 C129 C1          pop     bc
178 C12A E5          push    hl
179 C12B 2A23C1      ld      hl,(td1+1)
180 C12E 7C          ld      a,h
181 C12F FEC0        cp      0C0H
182 C131 D242C1      jp      nc,quit2
183 C134 210040      ld      hl,4000H
184 C137 09          add     hl,bc
185 C138 E5          push    hl
186 C139 C1          pop     bc
187 C13A E1          pop     hl
188 C13B 111000      ld      de,10H
189 C13E 19          add     hl,de
190 C13F C3EFC0      jp      loop4
191                  ;
192 C142 E1          quit2:  pop     hl
193 C143 F1          pop     af
194 C144 C9          ret
195                  ;
196 C145 0000      xpos:    defw    0
197 C147 00          ypos:    defb    0
198                  ;
199 C148 00002800    table:   defw    0,40,80,120,160,200,240,280,320,360
      C14C 50007800
      C150 A000C800
      C154 F0001801
      C158 40016801
200 C15C 9001B801      defw    400,440,480,520,560,600,640,680,720
      C160 E0010802
      C164 30025802
      C168 8002A802
      C16C D002
201 C16E F8022003      defw    760,800,840,880,920,960
      C172 48037003
      C176 9803C003
202                  ;
203 C17A 000A3F3F    pdata:   defb    00H,0AH,3FH,3FH,3FH,3DH,3DH,3FH
      C17E 3F3D3D3F
204 C182 3F3C3C3F      defb    3FH,3CH,3CH,3FH,3FH,3FH,3FH,00H
      C186 3F3F3F00
205 C18A 00A0FCFC      defb    00H,0A0H,0FCH,0FCH,0FCH,0BCH,0BCH,0FCH
      C18E FCBCBCFC
206 C192 FC3C3CFC      defb    0FCH,3CH,3CH,0FCH,0FCH,0FCH,0FCH,00H

```

つづく



リスト1-8 つづき

	C196	FCFCFC00			
207			:		
208	C19A	073066CE	defb	07H,30H,66H,0CEH,0CEH,4CH,0CH,0EH	
	C19E	CE4C0C0E			
209	C1A2	46DBDB80	defb	46H,0DBH,0DBH,80H,00H,00H,00H,0F9H	
	C1A6	000000F9			
210	C1AA	E00C6673	defb	0E0H,0CH,66H,73H,73H,32H,30H,70H	
	C1AE	73323070			
211	C1B2	62DBDB01	defb	62H,0DBH,0DBH,01H,00H,00H,00H,9FH	
	C1B6	0000009F			
212			:		
213	C1BA	073F66CE	defb	07H,3FH,66H,0CEH,0CEH,4CH,0CH,0EH	
	C1BE	CE4C0C0E			
214	C1C2	46C3C380	defb	46H,0C3H,0C3H,80H,00H,00H,00H,0F8H	
	C1C6	000000F8			
215	C1CA	E0FC6673	defb	0E0H,0FCH,66H,73H,73H,32H,30H,70H	
	C1CE	73323070			
216	C1D2	62C3C301	defb	62H,0C3H,0C3H,01H,00H,00H,00H,1FH	
	C1D6	0000001F			

## 〔2〕完全な重ね合わせ

これは、ちょうど背景にパターンをはめこむようなものです。図1-5をみてください。まず、RGBのパターン・データのORを取り、その“1”と“0”を反転します。こうして、得られたデータの“1”の部分は、パターンの外枠になります。

つぎに、このデータと背景のデータのANDを取ります。そして、さらにパターン・データとのORをとったものをVRAMに書き込みます（リスト1-9）。これで、背景にもパターンにも影響を与えない重ね合わせができます。しかし、その反面、速度が遅くなるという欠点を持っています。反転データは、最初から持っても良いのですが、ここではcpl命令で作っています。

## 〔3〕背景と移動物のVRAMを分離する

VRAMは、R・G・Bの3プレーンが独立しています。そこで、背景をVRAM1枚で表現して残りの2枚で移動物を描くというような方法が考えられます（もちろん逆も可）。

図1-6のパレット番号0と1の部分のREDとGREENを注目してください。両方とも“0”になっています。これは、REDとGREENに関係なくBLUEによってのみ色を支配できるということを意味しています。そこで、ここではこのBLUEのプレーン（パレット0と1）だけを背景に割り当てて2色で表現します。

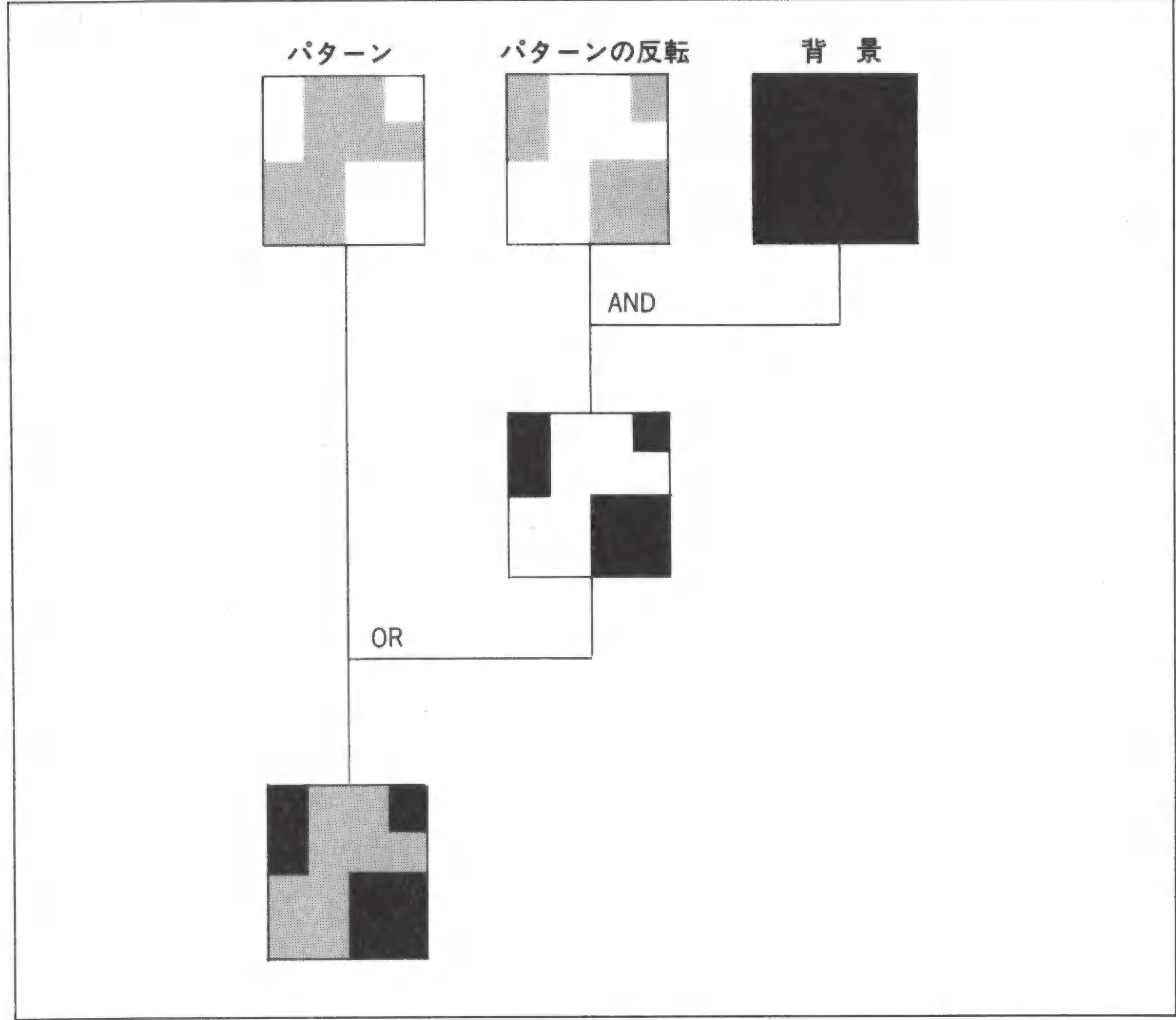


RED と GREEN の VRAM にパターンを書き込むわけですが、パレット 2 と 3， 4 と 5， 6 と 7 は同じ色にしなければなりません。なぜなら、背景のある無し、つまり BLUE のプレーンによって、パターンの色が変わるのを防ぐためです。

これで、重ね合わせをまったく意識せずにパターンの表示ができます。例をリスト 1-10 に示しますが、パターンの消去は単純に“0”を書き込むことで行っています。

この方式の長所は高速だということですが、短所としては移動物に 4 色，背景に 2 色しか色数が使えないことです。

図1-5 重ね合わせの例



リスト1-9

1	:	---	pattern move (and or) ---
2	:		
3	:		
4 C000	:	org	0C000H
5	:		

つづく



リストI-9

つづき

6	C000	CD61C1	call	rev	
7	C003	CD8DC0	call	erase	
8					
9					
10	C006	010019	key:	ld	bc,1900H
11	C009	ED78		in	a,(c)
12					:bc=i/o port address
13	C00B	FE32			:get key (a=ascii code)
14	C00D	CA25C0			
15	C010	FE34			
16	C012	CA3AC0			
17	C015	FE36			
18	C017	CA4EC0			
19	C01A	FE38			
20	C01C	CA63C0			
21	C01F	FE2A			
22	C021	C8			
23	C022	C306C0			
24					
25	C025	3A7EC1	down:	ld	a,(ypos)
26	C028	FE17		cp	23
27	C02A	D206C0		jp	nc,key
28	C02D	CD8DC0		call	erase
29	C030	3C		inc	a
30	C031	327EC1		ld	(ypos),a
31	C034	CDDFC0		call	disp
32	C037	C306C0		jp	key
33					
34	C03A	3A7CC1	left:	ld	a,(xpos)
35	C03D	B7		or	a
36	C03E	CA06C0		jp	z,key
37	C041	CD8DC0		call	erase
38	C044	3D		dec	a
39	C045	327CC1		ld	(xpos),a
40	C048	CDDFC0		call	disp
41	C04B	C306C0		jp	key
42					
43	C04E	3A7CC1	right:	ld	a,(xpos)
44	C051	FE26		cp	38
45	C053	D206C0		jp	nc,key
46	C056	CD8DC0		call	erase
47	C059	3C		inc	a
48	C05A	327CC1		ld	(xpos),a
49	C05D	CDDFC0		call	disp
50	C060	C306C0		jp	key
51					
52	C063	3A7EC1	up:	ld	a,(ypos)
53	C066	B7		or	a
54	C067	CA06C0		jp	z,key
55	C06A	CD8DC0		call	erase
56	C06D	3D		dec	a
57	C06E	327EC1		ld	(ypos),a
58	C071	CDDFC0		call	disp
59	C074	C306C0		jp	key
60					
61	C077	0180C1	adr:	ld	bc,table
62	C07A	2A7EC1		ld	hl,(ypos)
63	C07D	29		add	hl,hl
64	C07E	09		add	hl,bc
65	C07F	5E		ld	e,(hl)
66	C080	23		inc	hl
67	C081	56		ld	d,(hl)
68	C082	2A7CC1		ld	hl,(xpos)
69	C085	19		add	hl,de
70	C086	110040		ld	de,4000H
71	C089	19		add	hl,de
72	C08A	E5		push	hl
73	C08B	C1		pop	bc
74	C08C	C9		ret	
75					
76	C08D	F5	erase:	push	af
77	C08E	CD77C0		call	adr
78	C091	DD2132C2		ld	ix,bwork
79	C095	C5	loop1:	push	bc
80	C096	212800		ld	hl,28H
81	C099	09		add	hl,bc

つづ



## リストI-9

つづき

```

82 C09A 22C0C0      ld      (td+1),hl
83 C09D 3E02        ld      a,2
84 C09F F5          loop2:  push   af
85 C0A0 3E08        ld      a,8
86 C0A2 F5          loop3:  push   af
87 C0A3 DD7E00      ld      a,(ix)
88 C0A6 ED79        out      (c),a
89 C0A8 03          inc      bc
90 C0A9 DD23        inc      ix
91 C0AB DD7E00      ld      a,(ix)
92 C0AE ED79        out      (c),a
93 C0B0 0B          dec      bc
94 C0B1 DD23        inc      ix
95 C0B3 210008      ld      hl,0800H
96 C0B6 09          add      hl,bc
97 C0B7 44          ld      b,h
98 C0B8 4D          ld      c,l
99 C0B9 F1          pop      af
100 C0BA 3D          dec      a
101 C0BB C2A2C0      jp      nz,loop3
102 C0BE F1          pop      af
103 C0BF 010000      td:      ld      bc,0000H
104 C0C2 3D          dec      a
105 C0C3 C29FC0      jp      nz,loop2
106 C0C6 C1          pop      bc
107 C0C7 2AC0C0      ld      hl,(td+1)
108 C0CA 7C          ld      a,h
109 C0CB FEC0        cp      0C0H
110 C0CD D2DDC0      jp      nc,quit1
111 C0D0 210040      ld      hl,4000H
112 C0D3 09          add      hl,bc
113 C0D4 E5          push    hl
114 C0D5 C1          pop      bc
115 C0D6 111000      ld      de,10H
116 C0D9 19          add      hl,de
117 C0DA C395C0      jp      loop1
118 ;
119 C0DD F1          quit1: pop     af
120 C0DE C9          ret
121 ;
122 C0DF F5          disp:  push    af
123 C0E0 CD77C0      call   adr
124 C0E3 21B2C1      ld      hl,pdata
125 C0E6 DD2132C2    ld      ix,bwork      ;buck ground work area
126 C0EA C5          loop4:  push    bc
127 C0EB E5          push    hl
128 C0EC 111000      ld      de,0010H
129 C0EF 19          add      hl,de
130 C0F0 EB          ex      de,hl
131 C0F1 212800      ld      hl,0028H
132 C0F4 09          add      hl,bc
133 C0F5 223FC1      ld      (tdl+1),hl
134 C0F8 E1          pop      hl
135 C0F9 FD2112C2    ld      iy,revdata    ;reverse pattern data in iy
136 ;
137 C0FD 3E02        ld      a,2
138 C0FF F5          loop5:  push    af
139 C100 3E08        ld      a,8
140 C102 F5          loop6:  push    af
141 C103 ED78        in      a,(c)        ;read buck ground
142 C105 DD7700      ld      (ix),a        ;store bwork
143 C108 FDA600      and     (iy)          ;(buck ground) and (revdata)
144 C10B B6          or      (hl)        ;or pdata
145 C10C ED79        out     (c),a        ;print
146 C10E 03          inc     bc          ;increment address of vram
147 C10F DD23        inc     ix          ;increment address of bwork
148 C111 D5          push    de
149 C112 FDE5        push    iy          ;save revdata
150 C114 111000      ld      de,10H
151 C117 FD19        add     iy,de        ;revdata+16 in iy
152 C119 ED78        in      a,(c)
153 C11B DD7700      ld      (ix),a
154 C11E FDA600      and     (iy)
155 C121 FDE1        pop     iy          ;reload revdata
156 C123 D1          pop     de
157 C124 EB          ex      de,hl
158 C125 B6          or      (hl)

```

つづく



リストI-9 つづき

159	C126	EB		ex	de,hl	
160	C127	ED79		out	(c),a	
161	C129	0B		dec	bc	
162	C12A	23		inc	hl	
163	C12B	13		inc	de	
164	C12C	DD23		inc	ix	
165	C12E	FD23		inc	iy	
166	C130	E5		push	hl	
167	C131	210008		ld	hl,0800H	
168	C134	09		add	hl,bc	
169	C135	44		ld	b,h	
170	C136	4D		ld	c,l	
171	C137	E1		pop	hl	
172	C138	F1		pop	af	
173	C139	3D		dec	a	
174	C13A	C202C1		jp	nz,loop6	
175	C13D	F1		pop	af	
176	C13E	010000	td1:	ld	bc,0000H	
177	C141	3D		dec	a	
178	C142	C2FFC0		jp	nz,loop5	
179	C145	C1		pop	bc	
180	C146	E5		push	hl	
181	C147	2A3FC1		ld	hl,(td1+1)	
182	C14A	7C		ld	a,h	
183	C14B	FEC0		cp	0C0H	
184	C14D	D25EC1		jp	nc,quit2	
185	C150	210040		ld	hl,4000H	
186	C153	09		add	hl,bc	
187	C154	E5		push	hl	
188	C155	C1		pop	bc	
189	C156	E1		pop	hl	
190	C157	111000		ld	de,10H	
191	C15A	19		add	hl,de	
192	C15B	C3EAC0		jp	loop4	
193			:			
194	C15E	E1	quit2:	pop	hl	
195	C15F	F1		pop	af	
196	C160	C9		ret		
197			:			
198	C161	21B2C1	rev:	ld	hl,pdata	:address of pattern data in hl
199	C164	1112C2		ld	de,revdata	:address of reverse data in de
200	C167	0620		ld	b,32	:set counter
201	C169	E5	jmp1:	push	hl	
202	C16A	D5		push	de	
203	C16B	7E		ld	a,(hl)	:pattern data in a
204	C16C	112000		ld	de,32	
205	C16F	19		add	hl,de	:pdata+32
206	C170	B6		or	(hl)	:blue or red in a
207	C171	19		add	hl,de	:pdata+64
208	C172	B6		or	(hl)	
209	C173	2F		cpl		:reverse
210	C174	D1		pop	de	
211	C175	12		ld	(de),a	:store reverse data
212	C176	E1		pop	hl	
213	C177	13		inc	de	
214	C178	23		inc	hl	
215	C179	10EE		djnz	jmp1	
216	C17B	C9		ret		
217			:			
218	C17C	0000	xpos:	defw	0	
219	C17E	0000	ypos:	defw	0	
220			:			
221	C180	00002800	table:	defw	0,40,80,120,160,200,240,280,320,360	
	C184	50007800				
	C188	A000C800				
	C18C	F0001801				
	C190	40016801				
222	C194	9001B801		defw	400,440,480,520,560,600,640,680,720	
	C198	E0010802				
	C19C	30025802				
	C1A0	8002A802				
	C1A4	D002				
223	C1A6	F8022003		defw	760,800,840,880,920,960	
	C1AA	48037003				
	C1AE	9803C003				
224			:			

つづく



リストI-9 つづき

```
225 C1B2 000A3F3F pdata: defb 00H,0AH,3FH,3FH,3FH,3DH,3DH,3FH
    C1B6 3F3D3D3F          defb 3FH,3CH,3CH,3FH,3FH,3FH,3FH,00H
226 C1BA 3F3C3C3F          defb
    C1BE 3F3F3F00          defb 00H,0A0H,0FCH,0FCH,0FCH,0BCH,0BCH,0FCH
227 C1C2 00A0FCFC          defb
    C1C6 FCBCBCFC          defb 0FCH,3CH,3CH,0FCH,0FCH,0FCH,0FCH,00H
228 C1CA FC3C3CFC          defb
    C1CE FCFCFC00          ;
229                                defb 07H,30H,66H,0CEH,0CEH,4CH,0CH,0EH
230 C1D2 073066CE          defb
    C1D6 CE4C0C0E          defb 46H,0DBH,0DBH,80H,00H,00H,00H,0F9H
231 C1DA 46DBDB80          defb
    C1DE 000000F9          defb 0E0H,0CH,66H,73H,73H,32H,30H,70H
232 C1E2 E00C6673          defb
    C1E6 73323070          defb 62H,0DBH,0DBH,01H,00H,00H,00H,9FH
233 C1EA 62DBDB01          defb
    C1EE 0000009F          ;
234                                defb 07H,3FH,66H,0CEH,0CEH,4CH,0CH,0EH
235 C1F2 073F66CE          defb
    C1F6 CE4C0C0E          defb 46H,0C3H,0C3H,80H,00H,00H,00H,0F8H
236 C1FA 46C3C380          defb
    C1FE 000000F8          defb 0E0H,0FCH,66H,73H,73H,32H,30H,70H
237 C202 E0FC6673          defb
    C206 73323070          defb 62H,0C3H,0C3H,01H,00H,00H,00H,1FH
238 C20A 62C3C301          defb
    C20E 0000001F          ;
239                                revdata:defs 32
240 C212                                ;
241                                bwork: defs 96
242 C232
243
```

図I-6 パレット・コード

パレット番号	B	R	G
0	0	0	0
1	1	0	0
2	0	1	0
3	1	1	0
4	0	0	1
5	1	0	1
6	0	1	1
7	1	1	1

リストI-10

```
1                                ;
2                                ; --- separate vram ---
3                                ;
4 C000                                org 0C000H
5                                ;
6 C000 CD30C1                        call plt
7                                ;
8 C003 010019                        key: ld bc,1900H          ;bc=i/o port address
9 C006 ED78                          in a,(c)          ;get key (a=ascii code)
10                                ;
11 C008 FE32                          cp '2'          ;'2'
12 C00A CA22C0                        jp z,down          ;if a=2 then down
13 C00D FE34                          cp '4'          ;'4'
14 C00F CA3AC0                        jp z,left         ;if a=4 then left
```

つづく



リストI-10

つづき

15 C012 FE36	cp	'6'	;'6'
16 C014 CA51C0	jp	z,right	;if a=6 then right
17 C017 FE38	cp	'8'	;'8'
18 C019 CA69C0	jp	z,up	;if a=8 then up
19 C01C FE2A	cp	'*'	;'*'
20 C01E C8	ret	z	;return to system
21 C01F C303C0	jp	key	
22			
23 C022 3A44C1	down:	ld	a,(ypos)
24 C025 FE17		cp	23
25 C027 D203C0		jp	nc,key
26 C02A CD96C0		call	erase
27 C02D 3C		inc	a
28 C02E 3244C1		ld	(ypos),a
29 C031 2178C1		ld	hl,pdata
30 C034 CDDAC0		call	disp
31 C037 C303C0		jp	key
32			
33 C03A 3A42C1	left:	ld	a,(xpos)
34 C03D B7		or	a
35 C03E CA03C0		jp	z,key
36 C041 CD96C0		call	erase
37 C044 3D		dec	a
38 C045 3242C1		ld	(xpos),a
39 C048 2178C1		ld	hl,pdata
40 C04B CDDAC0		call	disp
41 C04E C303C0		jp	key
42			
43 C051 3A42C1	right:	ld	a,(xpos)
44 C054 FE26		cp	38
45 C056 D203C0		jp	nc,key
46 C059 CD96C0		call	erase
47 C05C 3C		inc	a
48 C05D 3242C1		ld	(xpos),a
49 C060 2178C1		ld	hl,pdata
50 C063 CDDAC0		call	disp
51 C066 C303C0		jp	key
52			
53 C069 3A44C1	up:	ld	a,(ypos)
54 C06C B7		or	a
55 C06D CA03C0		jp	z,key
56 C070 CD96C0		call	erase
57 C073 3D		dec	a
58 C074 3244C1		ld	(ypos),a
59 C077 2178C1		ld	hl,pdata
60 C07A CDDAC0		call	disp
61 C07D C303C0		jp	key
62			
63 C080 0146C1	adr:	ld	bc,table
64 C083 2A44C1		ld	hl,(ypos)
65 C086 29		add	hl,hl
66 C087 09		add	hl,bc
67 C088 5E		ld	e,(hl)
68 C089 23		inc	hl
69 C08A 56		ld	d,(hl)
70 C08B 2A42C1		ld	hl,(xpos)
71 C08E 19		add	hl,de
72 C08F 110080		ld	de,0000H
73 C092 19		add	hl,de
74 C093 44		ld	b,h
75 C094 4D		ld	c,l
76 C095 C9		ret	
77			
78 C096 F5	erase:	push	af
79 C097 CD80C0		call	adr
80 C09A C5	loop1:	push	bc
81 C09B 212800		ld	hl,28H
82 C09E 09		add	hl,bc
83 C09F 22BCC0		ld	(td+1),hl
84 C0A2 3E02		ld	a,2
85 C0A4 F5	loop2:	push	af
86 C0A5 3E08		ld	a,8
87 C0A7 F5	loop3:	push	af
88 C0A8 AF		xor	a
89 C0A9 ED79		out	(c),a
90 C0AB 03		inc	bc
91 C0AC ED79		out	(c),a
92 C0AE 0B		dec	bc
93 C0AF 210008		ld	hl,0000H
94 C0B2 09		add	hl,bc
95 C0B3 44		ld	b,h

つづく



## リストI-10

つづき

```

96 C0B4 4D      ld      c,l          ;address of next line
97 C0B5 F1      pop      af
98 C0B6 3D      dec      a          ;counter-1
99 C0B7 C2A7C0  jp      nz,loop3
100 C0BA F1      pop      af
101 C0BB 010000  td:    ld      bc,0000H
102 C0BE 3D      dec      a          ;counter-1
103 C0BF C2A4C0  jp      nz,loop2
104 C0C2 C1      pop      bc
105 C0C3 2ABCC0  ld      hl,(td+1)      ;address of vram+28h in hl
106 C0C6 CB74    bit      6,h
107 C0C8 C2D8C0  jp      nz,quit1
108 C0CB 210040  ld      hl,4000H
109 C0CE 09      add     hl,bc
110 C0CF 44      ld      b,h
111 C0D0 4D      ld      c,l          ;next base address in bc
112 C0D1 111000  ld      de,10H
113 C0D4 19      add     hl,de
114 C0D5 C39AC0  jp      loop1
115              ;
116 C0D8 F1      quit1: pop     af
117 C0D9 C9      ret
118              ;
119 C0DA F5      disp:  push    af
120 C0DB CD80C0  call    adr
121 C0DE 2178C1  ld      hl,pdata
122 C0E1 C5      loop4:  push    bc
123 C0E2 E5      push    hl
124 C0E3 111000  ld      de,0010H
125 C0E6 19      add     hl,de
126 C0E7 EB      ex      de,hl
127 C0E8 212800  ld      hl,0028H
128 C0EB 09      add     hl,bc
129 C0EC 220FC1  ld      (td1+1),hl
130 C0EF E1      pop     hl
131              ;
132 C0F0 3E02      ld      a,2
133 C0F2 F5      loop5:  push    af
134 C0F3 3E08      ld      a,8
135 C0F5 F5      loop6:  push    af
136 C0F6 7E      ld      a,(hl)      ;load pattern data
137 C0F7 ED79      out     (c),a      ;print pattern data
138 C0F9 03      inc     bc
139 C0FA 1A      ld      a,(de)      ;load pattern data
140 C0FB ED79      out     (c),a
141 C0FD 0B      dec     bc
142 C0FE 23      inc     hl          ;increment address of pattern data
143 C0FF 13      inc     de
144 C100 E5      push    hl
145 C101 210008  ld      hl,0800H
146 C104 09      add     hl,bc
147 C105 44      ld      b,h
148 C106 4D      ld      c,l
149 C107 E1      pop     hl
150 C108 F1      pop     af
151 C109 3D      dec     a
152 C10A C2F5C0  jp      nz,loop6
153 C10D F1      pop     af
154 C10E 010000  td1:    ld      bc,0000H
155 C111 3D      dec     a
156 C112 C2F2C0  jp      nz,loop5
157 C115 C1      pop     bc
158 C116 E5      push    hl
159 C117 2A0FC1  ld      hl,(td1+1)
160 C11A CB74    bit     6,h
161 C11C C22DC1  jp      nz,quit2
162 C11F 210040  ld      hl,4000H
163 C122 09      add     hl,bc
164 C123 E5      push    hl
165 C124 C1      pop     bc
166 C125 E1      pop     hl
167 C126 111000  ld      de,10H
168 C129 19      add     hl,de
169 C12A C3E1C0  jp      loop4
170              ;
171 C12D E1      quit2: pop     hl
172 C12E F1      pop     af
173 C12F C9      ret
174              ;
175 C130 010010  plt:    ld      bc,1000H      ;set palet (blue)
176 C133 3E02      ld      a,02H

```

つづく



リスト1-10 つづき

```

177 C135 ED79      out    (c),a
178 C137 04        inc    b                ;set palet (red)
179 C138 3ECC      ld     a,0CCH
180 C13A ED79      out    (c),a
181 C13C 04        inc    b                ;set palet (green)
182 C13D 3EF0      ld     a,0F0H
183 C13F ED79      out    (c),a
184 C141 C9        ret
185                ;
186 C142 0000      xpos:   defw    0
187 C144 0000      ypos:   defw    0
188                ;
189 C146 00002800   table:  defw    0,40,80,120,160,200,240,280,320,360
      C14A 50007800
      C14E A000C800
      C152 F0001801
      C156 40016801
190 C15A 9001B801      defw    400,440,480,520,560,600,640,680,720
      C15E E0010802
      C162 30025802
      C166 8002A802
      C16A D002
191 C16C F8022003      defw    760,800,840,880,920,960
      C170 48037003
      C174 9803C003
192                ;
193 C178 01010101   pdata:  defb    01H,01H,01H,01H,01H,21H,33H,23H
      C17C 01213323
194 C180 26272667      defb    26H,27H,26H,67H,66H,66H,0E2H,0C0H
      C184 6666E2C0
195 C188 00800080      defb    00H,80H,00H,80H,00H,88H,0CH,0CH
      C18C 00880C0C
196 C190 8808880A      defb    88H,08H,88H,0AH,8AH,4AH,4BH,03H
      C194 8A4A4B03
197                ;
198 C198 01000100      defb    01H,00H,01H,00H,01H,30H,03H,33H
      C19C 01300333
199 C1A0 37B7BFBF      defb    37H,0B7H,0BFH,0BFH,0BFH,0BEH,32H,0C0H
      C1A4 BFBF32C0
200 C1A8 80008000      defb    80H,00H,80H,00H,80H,0CH,0C0H,0CCH
      C1AC 800CC0CC
201 C1B0 ECEDFDFD      defb    0ECH,0EDH,0FDH,0FDH,0FDH,7DH,4CH,03H
      C1B4 FD7D4C03

```

## 1-3-2 PCGによるスクロール

ここではテキスト画面の逆スクロールだけを取りあげます。逆スクロールの原理は図1-7に示すとおりです。リスト1-11は'mstart'と'mend'にそれぞれマップのスタート・アドレスとエンド・アドレスをいれてコールします。マップには、40文字分を1ラインとして、ASCIIコードを入れてください。







リストI-II

```

1      ;
2      ;      --- SCROLL (PCG) ---
3      ;
4 C000      org      0C000H
5      ;
6 C000 2A47C0      ld      hl,(mstart)      ;start address of the map
7 C003 E5          loop1: push    hl          ;save start address of the map
8 C004 01BF33      ld      bc,33BFH         ;get address of vram (x=39:y=23)
9 C007 21E733      ld      hl,33E7H         ;put address of vram (x=39:y=24)
10     ;
11 C00A 1E19      ld      e,25              ;y colume in e (counter)
12 C00C 1628      loop2: ld      d,40        ;x colume in d (counter)
13 C00E C5        loop3: push    bc          ;save address of vram to bc
14 C00F ED78      in      a,(c)             ;ascii code in a
15 C011 44        ld      b,h              ;print address in bc
16 C012 4D        ld      c,l
17 C013 ED79      out     (c),a             ;print
18 C015 CBA0      res      4,b              ;attribute address
19 C017 3E27      ld      a,27H            ;attribute code in a (normal,ramcg)
20 C019 ED79      out     (c),a             ;print
21 C01B C1        pop     bc               ;load address of map
22 C01C 0B        dec     bc               ;print address-1
23 C01D 2B        dec     hl               ;load address-1
24 C01E 15        dec     d                ;x colume-1      in d
25 C01F 20ED      jr      nz,loop3          ;if x<>0 then loop3
26 C021 1D        dec     e                ;y colume-1      in e
27 C022 20E8      jr      nz,loop2          ;if y<>0 then loop2
28     ;
29 C024 E1        pop     hl               ;load put address of vram to hl
30 C025 010030     ld      bc,3000H           ;base address of vram
31 C028 1628      ld      d,40            ;x colume in d (counter)
32 C02A 7E        loop4: ld      a,(hl)      ;(put address of vram) in a
33 C02B ED79      out     (c),a             ;print
34 C02D CBA0      res      4,b              ;attribute address
35 C02F 3E27      ld      a,27H            ;attribute code in a (normal,ramcg)
36 C031 ED79      out     (c),a             ;print
37 C033 CBE0      set     4,b              ;text address
38 C035 23        inc     hl               ;put address of vram+1
39 C036 03        inc     bc               ;get address of vram+1
40 C037 15        dec     d                ;x colume-1
41 C038 20F0      jr      nz,loop4          ;if x<>0 then loop4
42     ;
43 C03A ED5B49C0     ld      de,(mend)           ;high address of end of the map
44 C03E 7A        ld      a,d
45 C03F 94        sub     h
46 C040 20C1      jr      nz,loop1
47 C042 7B        ld      a,e
48 C043 95        sub     l
49 C044 20BD      jr      nz,loop1          ;if a<>0 then loop1
50 C046 C9        ret
51     ;
52 C047 00D0      mstart: defw    0D000H
53 C049 00F0      mend:   defw    0F000H

```



# 第2章 グラフィックス

---

## 2-1 PSET

## 2-2 LINEとCIRCLE

## 2-3 PAINT

---

本章では、BASICにあるようなLINEやPAINTを作ってみます。これらは、プログラムそのものよりもアルゴリズムが重要なので、BASICで記述して例を示し、それをマシン語化するようにします。なお、簡単なためグラフィックのモードは640×200ドットに固定します。



## 2-1

## PSET

PSET はグラフィックの最も基本的なものであり、最もハードウェアに近いものと言えます。したがって、このルーチンさえあれば、後述する LINE や CIRCLE など了他機種へ移植するのは簡単なことです。また、CP / M 上の高級言語でもグラフィックを使えるようになります。

### 2-1-1 アルゴリズム

PSET ルーチンでは、まず始めに絶対座標から実際に書き込む VRAM のアドレスを求めなければなりません。ところが、X 1 は PC-8801 や FM-7 と VRAM の並びかたが異なり、この変換はこれらのマシンより少々複雑です。

まず、与えられた X, Y から  $\text{INT}(X/8)$ ,  $\text{INT}(Y/8)$  としてキャラクタ座標を求めます。これより、左上端からのオフセット値は、

$$\text{INT}(X/8) + \text{INT}(Y/8) * 80$$

で求められます。この式は図 2-1 の斜線部のようなアドレスを求めるものです。後は Y 方向に &H 800 ずつ増加しているので、アドレスを求める式は次のようになります。

$$\text{INT}(X/8) + \text{INT}(Y/8) * 80 + (Y \text{ MOD } 8) * \&H800$$

この式から求まった値に &H 4000 を加えると BLUE の VRAM, さらに &H 4000 を加えると RED の VRAM, さらに加えると GREEN の VRAM のアドレスになります。この式が正しいかどうかは、リスト 2-1 で試してみてください。



## 図2-1 VRAMのアドレス

640×200ドット(値は16進表示)

8ドット

0000	0001	0002
0800	0801	0802
1000	1001	1002
1800	1801	1802
2000	2001	2002
2800	2801	2802
3000	3001	3002
3800	3801	3802
0050	0051	6052
0850	0851	0852
1050	1051	1052

8ドット

004E	004F
084E	084F
104E	104F
184E	184F
204E	204F
284E	284F
304E	304F
384E	384F
009E	009F
089E	089F
109E	109F

3780	3781	3782
3F80	3F81	3F82

37CE	37CF
3FCE	3FCF

リスト2-1

```

1000 INIT : WIDTH 80 : CLS 4
1010 DEFINT A-Z
1020 :
1030 LABEL "LOOP"
1040     INPUT "X0,Y0";X0,Y0
1050     IF (X0 > 639 OR X0 < 0) OR (Y0 > 199 OR Y0 < 0)
        THEN PRINT "Error !" : GOTO "LOOP"
1060     VRAMADR=INT(X0/8)+INT(Y0/8)*80+(Y0 MOD 8)*&H800
1070     OUT VRAMADR+&H4000,&HFF           : ' BLUE
1080     OUT VRAMADR+&H8000,0              : ' RED
1090     OUT VRAMADR+&HC000,0              : ' GREEN
1100 GOTO "LOOP"

```

## 2-1-2 プログラム

絶対座標から VRAM アドレスを求めるプログラムは、一見複雑そうですが、以外に簡単にマシン語にできます。そのプログラムがリスト 2-2 です。bc レジスタに X 座標, de



レジスタに Y 座標を入れて'calcva' をコールすると hl レジスタに VRAM アドレスを入れて返ってきます。

リスト 2-2

```
1      ;
2      ; --- calc. vram address ---
3      ;      entry          bc := x
4      ;      de := y
5      ;      returns       hl <- vram address
6      ;      modifies      af, bc, de, hl
7      ;
8 C000      org      0C000H
9
10 C000 CB38      calcva: srl      b
11 C002 CB19      rr      c
12 C004 CB38      srl      b
13 C006 CB19      rr      c
14 C008 CB38      srl      b
15 C00A CB19      rr      c          ; bc:=int(x0/8)
16
17 C00C 7B      ld      a,e
18 C00D F5      push     af
19 C00E E6F8      and      0F8H
20 C010 5F      ld      e,a          ; de:=int(y0/8)*8
21 C011 F1      pop      af
22 C012 E607      and      7          ; a:=y0 mod 8
23 C014 EB      ex      de,hl
24 C015 29      add      hl,hl          ; *2
25 C016 E5      push     hl
26 C017 29      add      hl,hl          ; *4
27 C018 29      add      hl,hl          ; *8
28 C019 D1      pop      de
29 C01A 19      add      hl,de          ; *10
30      ;          ; hl:=int(y0/8)*80
31 C01B 87      add      a,a
32 C01C 87      add      a,a
33 C01D 87      add      a,a
34 C01E 57      ld      d,a
35 C01F 1E00      ld      e,0          ; de:=(y0 mod 8)*8h800
36 C021 19      add      hl,de
37 C022 09      add      hl,bc          ; hl:=int(x0/8)+int(y0/8)*80
38      ;          ; +(y0 mod 8)*8h800
39 C023 C9      ret
```

リスト 2-3 は PSET ルーチンの本体ですが，ここではビットを ON または OFF するためにプログラム自身を書き換えるという姑息（？）なテクニックを使っています。ビットの ON, OFF には SET, RES 命令を使っていますが，これらの OP コードは図 2-2 のようになっています。ただし，この図は SET, RES 命令の一部です。ここでは e レジスタに VRAM のデータを入れているので，このレジスタに対してビットのセット，リセットを行います。ここで何ビット目を操作するかという，これは次のように X 座標から求められます。



## X MOD 8

## リスト2-3

```

1      ;
2      ;      ---      subroutine 'pset' ---
3      ;      entry      (x0)      := point x
4      ;      (y0)      := point y
5      ;      (color) := dot color
6      ;      modifies   af,bc,de,hl
7      ;
8      C000      calcva equ      0C000H
9      ;
10     C100      org      0C100H
11     C100 C308C1      jp      pset
12     ;
13     C103 0000      x0:      defw      0000H
14     C105 0000      y0:      defw      0000H
15     C107 06        color: defb      6
16     ;
17     ;
18     C108      pset      equ      $
19     ;
20     C108 ED4B03C1   pset1: ld      bc,(x0)
21     C10C ED5B05C1   ld      de,(y0)
22     C110 217F02     pset2: ld      hl,639
23     C113 B7         or      a
24     C114 ED42       sbc      hl,bc
25     C116 D8         ret      c
26     C117 21C700     ld      hl,199
27     C11A ED52       sbc      hl,de
28     C11C D8         ret      c
29     ;
30     C11D 79         pset3: ld      a,c
31     C11E 2F         cpl
32     C11F E607       and      7          ; a:=cpl(x0 mod 8)
33     C121 327BC1     ld      (xmod8),a
34     C124 CD00C0     call     calcva
35     C127 3A7BC1     ld      a,(xmod8)
36     C12A 87         add      a,a          ; shift left 3 bits
37     C12B 87         add      a,a
38     C12C 87         add      a,a
39     C12D F5         push     af
40     C12E F6C3       or      0C3H          ; or 11000011b
41     C130 3272C1     ld      (@set),a      ;
42     C133 F1         pop      af
43     C134 F683       or      83H          ; or 10000011b
44     C136 3277C1     ld      (@reset),a
45     ;
46     C139 110040     ld      de,4000H      ; blue
47     C13C 19         add      hl,de
48     C13D 44         ld      b,h
49     C13E 4D         ld      c,l
50     C13F ED58       in      e,(c)
51     C141 CD58C1     call     blue
52     C144 110040     ld      de,4000H      ; red
53     C147 19         add      hl,de
54     C148 44         ld      b,h
55     C149 ED58       in      e,(c)
56     C14B CD61C1     call     red
57     C14E 110040     ld      de,4000H      ; green
58     C151 19         add      hl,de

```

つづく



リスト2-3 つづき

```

59 C152 44          ld      b,h
60 C153 ED58        in      e,(c)
61 C155 C36AC1      jp      green
62                  ;
63 C158 3A07C1      blue:   ld      a,(color)
64 C15B CB47        bit     0,a
65 C15D 2817        jr      z,resbit
66 C15F 1810        jr      setbit
67 C161 3A07C1      red:    ld      a,(color)
68 C164 CB4F        bit     1,a
69 C166 280E        jr      z,resbit
70 C168 1807        jr      setbit
71 C16A 3A07C1      green:  ld      a,(color)
72 C16D CB57        bit     2,a
73 C16F 2805        jr      z,resbit
74
75                  ;
76 C171 CB          setbit: defb    0CBH          ; set n,e
77 C172             @set:  defs    1
78 C173 ED59        out     (c),e
79 C175 C9          ret
80 C176 CB          resbit: defb    0CBH          ; res n,e
81 C177             @reset: defs    1
82 C178 ED59        out     (c),e
83 C17A C9          ret
84                  ;
85 C17B             xmod8:  defs    1
86                  ;

```

図2-2 SET, RES命令のOPコード

ニモニック		OPコード			
		76	543	210	16進
SET	b,r	11	001	011	CB
		11	b	r	—
RES	b,r	11	001	011	CB
		10	b	r	—

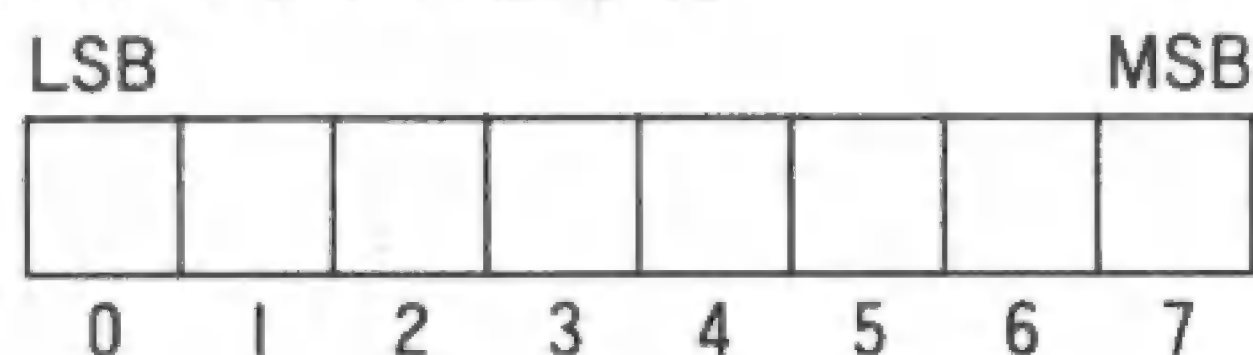
r	レジスタ
000	B
001	C
010	D
011	E
100	H
101	L
111	A

b	Bit
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7



ところが、VRAM のビットの並び方は図 2-3 のようになっており、図 2-2 のビット指定とは逆になっています。そのため、CPL 命令で 1 と 0 を反転しています。

図 2-3 VRAM のビットの並び方



このルーチンでは、最初に座標値のチェックを行っていますが、色コード（'color'）についてはチェックしていません。色コードは下位 3 ビットだけしか見ていないので、たとえば、81 H という値を入れても青のドットを打ちます。

この PSET ルーチンには、ワーク・エリアの x 0, y 0 に座標をセットして 'pset' または 'pset 1' をコールする、bc レジスタに X 座標、de レジスタに Y 座標を入れて 'pset 2' をコールする、さらに座標値のチェックをしない 'pset 3' をコールするという 3 種類のエントリーポイントがあります。後述する LINE, CIRCLE, PAINT ではそれぞれこのラベル名でコールします。



## 2-2

# LINEとCIRCLE

ここで、紹介する LINE は BASIC の LINE 命令とは異なり、ボックスを描いたり塗つぶしたりすることはできません。また、色もタイル・パターンではなく 1 色で指定します。CIRCLE も非常に単純なもので、楕円や円弧を描いたりする芸当はなく、単に円を描くだけです。

### 2-2-1 LINE

2 点の座標が与えられたときに、その間を結ぶ、つまり線分を引くにはいくつかの方法があります。いちばん簡単なのは、直線の方程式、

$$y = ax + b$$

の  $a$ 、 $b$  を

$$(y - y_1) = \frac{y_2 - y_1}{x_2 - x_1} (x - x_1)$$

から求め、 $x$  の値を  $x_1$  から  $x_2$  まで変化させ  $y$  の値を求めながら点を打っていくことです。ところが、これでは求まった値はふつう実数値となり、画面の座標のように整数値になることはめったにありません。このため、このアルゴリズムどおりにマシン語化するには実数演算が必要になり、大変面倒になります。

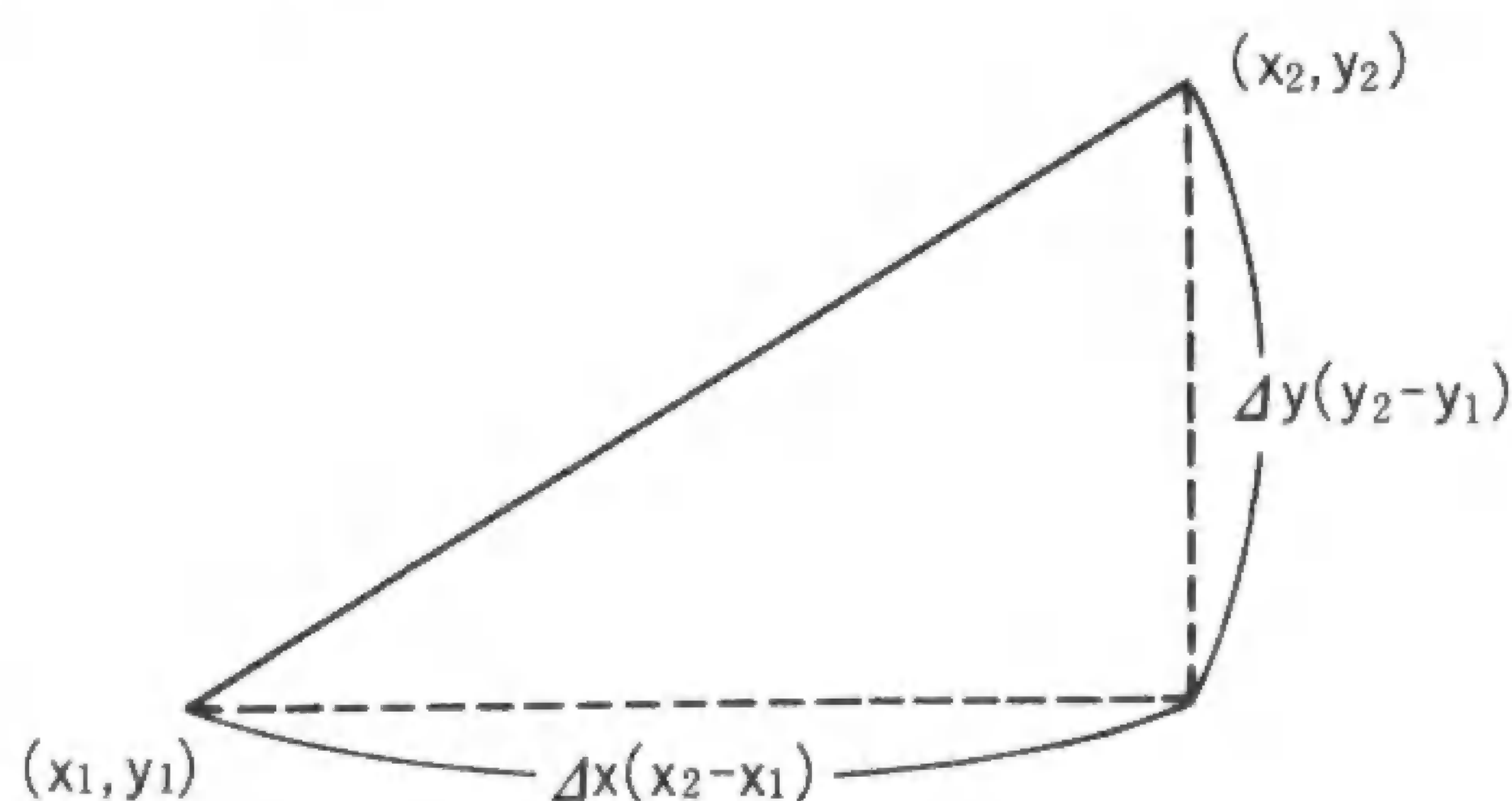
ここで使うアルゴリズムは DDA によるもので、これだと整数演算のみで LINE を引くことができます。DDA というのはデジタル微分解析法 (Digital Differential Analyze) の略で線形微分方程式を解くために微積分をアナログ計算機で行う基となったアルゴリズムです。DDA によるアルゴリズムには  $x$  方向、 $y$  方向のうち、大きいほうの座標は



1ずつ変化させ、もう一方の座標は真の値に最も近い整数値を選ぶ『単純形 DDA』と  $x$ 、 $y$  両方向に DDA 演算する『対称形 DDA』があります。ここでは前者の単純形 DDA を使います。

点  $(x_1, y_1)$  から点  $(x_2, y_2)$  へ線を引くことを考えます (図 2-4)。この図では、 $x_2 > x_1$ 、 $y_2 > y_1$  で、なおかつ  $\Delta x > \Delta y$  ですから、 $x$  については単純に 1 ずつ加えていきます。 $y$  座標の変化は次のようにします。

図2-4 線分の一例



まず、EE という変数を用意し、 $x$  座標が 1 増加するごとに  $\Delta y$  を加えます。これが  $\Delta x$  より大きくなったら  $y$  座標を 1 増加させ、EE から  $\Delta x$  を引きます。これを  $x_1 = x_2$  になるまで繰り返します。

これまでは、 $\Delta x > \Delta y$  の場合の説明でしたが、 $\Delta x < \Delta y$  の場合は、 $y$  について 1 ずつ増加させて  $x$  については同様の演算をして求めます。また、 $x_2 < x_1$  や  $y_2 < y_1$  の場合は、1 加えるところを 1 減じればよいでしょう。

以上、見てきたように単純形 DDA を使った LINE ルーチンでは、まず 2 点の座標から場合分けをします。BASIC で書いたプログラムをリスト 2-4 に、マシン語のプログラムをリスト 2-5 に示します。



# リスト2-4

```

1000 INIT : WIDTH 80 : CLS 4
1010 DEFINT A-Z
1020 :
1030 INPUT "XS,YS";XS,YS
1040 INPUT "XE,YE";XE,YE
1050 :
1060 DX=ABS(XE-XS) : DY=ABS(YE-YS)
1070 SX=SGN(XE-XS) : SY=SGN(YE-YS)
1080 IF DX < DY
    THEN SWAP XS,YS : SWAP XE,YE : SWAP DX,DY : SWAP SX,SY : FLAG=-1
1090 :
1100 EE=DX/2
1110 LABEL"LOOP"
1120     IF FLAG THEN PSET(YS,XS,7)
        ELSE PSET(XS,YS,7)
1130     IF XS=XE THEN END
1140     XS=XS+SX
1150     EE=EE+DY
1160     IF EE>=DX THEN EE=EE-DX : YS=YS+SY
1170 GOTO "LOOP"

```

# リスト2-5

```

1      :
2      :      ---      subroutine 'line' ---
3      :
4      :      entry      (xs)      := start point x
5      :                  (ys)      := start point y
6      :                  (xe)      := end point x
7      :                  (ye)      := end point y
8      :                  (color) := dot color
9      :                  modifies af, bc, de, hl
10     :
11 C108 pset      equ      0C108H
12 C103 x0        equ      0C103H
13 C10C y0        equ      0C10CH
14 C107 color     equ      0C107H
15     ;
16 C200                org      0C200H
17 C200 C316C2         jp       line
18     ;
19 C203 0000          xs:      defw      0
20 C205 0000          ys:      defw      0
21 C207 0000          xe:      defw      0
22 C209 0000          ye:      defw      0
23     ;
24 C20B 0000          dx:      defw      0
25 C20D 0000          dy:      defw      0
26 C20F 0000          ee:      defw      0
27 C211 0000          sx:      defw      0
28 C213 0000          sy:      defw      0
29 C215 00            fl:      defb      0
30     ;
31 C216 210100        line:    ld        hl, 1                ; sx:=1
32 C219 2211C2         ld        (sx), hl                ; sy:=1
33 C21C 2213C2         ld        (sy), hl
34 C21F AF            xor        a                        ; fl:=0
35 C220 3215C2         ld        (fl), a
36 C223 2A07C2         ld        hl, (xe)                ; dx:=abs(xe-xs)
37 C226 ED5B03C2       ld        de, (xs)
38 C22A ED52          sbc        hl, de
39 C22C 3007          jr         nc, line1                ; if xe<xs then
40 C22E 01FFFF         ld        bc, 0FFFFH                ;   sx:=-1
41 C231 ED4311C2       ld        (sx), bc
42 C235 CDF4C2        line1:  call      abs
43 C238 220BC2         ld        (dx), hl
44 C23B E5            push      hl
45 C23C 2A09C2         ld        hl, (ye)                ; dy:=abs(ye-ys)

```

つづく



## リスト2-5 つづき

46	C23F	ED5B05C2	ld	de,(ys)	
47	C243	ED52	sbc	hl,de	
48	C245	3007	jr	nc,line2	; if ye<ys then
49	C247	01FFFF	ld	bc,0FFFFH	; sy:=-1
50	C24A	ED4313C2	ld	(sy),bc	
51	C24E	CDF4C2	line2: call	abs	; hl:=abs(ye-ys)
52	C251	220DC2	ld	(dy),hl	
53	C254	D1	pop	de	; de:=abs(xe-xs)
54	C255	ED52	sbc	hl,de	
55	C257	383D	jr	c,l1	; if xd<y d then
56	C259	2A0BC2	ld	hl,(dx)	; swap dx,dy
57	C25C	ED5B0DC2	ld	de,(dy)	
58	C260	220DC2	ld	(dy),hl	
59	C263	ED530BC2	ld	(dx),de	
60	C267	2A03C2	ld	hl,(xs)	; swap xs,ys
61	C26A	ED5B05C2	ld	de,(ys)	
62	C26E	2205C2	ld	(ys),hl	
63	C271	ED5303C2	ld	(xs),de	
64	C275	2A07C2	ld	hl,(xe)	; swap xe,ye
65	C278	ED5B09C2	ld	de,(ye)	
66	C27C	2209C2	ld	(ye),hl	
67	C27F	ED5307C2	ld	(xe),de	
68	C283	2A11C2	ld	hl,(sx)	; swap sx,sy
69	C286	ED5B13C2	ld	de,(sy)	
70	C28A	2213C2	ld	(sy),hl	
71	C28D	ED5311C2	ld	(sx),de	
72	C291	3E01	ld	a,l	; fl:=1
73	C293	3215C2	ld	(fl),a	
74	C296	2A0BC2	11: ld	hl,(dx)	; ee:=dx/2
75	C299	CB2C	sra	h	
76	C29B	CB1D	rr	l	
77	C29D	220FC2	ld	(ee),hl	
78	C2A0	2A03C2	15: ld	hl,(xs)	
79	C2A3	ED5B05C2	ld	de,(ys)	
80	C2A7	3A15C2	ld	a,(fl)	; if fl<>0 then
81	C2AA	B7	or	a	; x0:=ys
82	C2AB	2801	jr	z,l2	; y0:=xs
83	C2AD	EB	ex	de,hl	; else
84	C2AE	2203C1	12: ld	(x0),hl	; x0:=xs
85	C2B1	ED530CC1	ld	(y0),de	; y0:=ys
86	C2B5	CD08C1	call	pset	; call pset (x0,y0)
87	C2B8	2A03C2	ld	hl,(xs)	
88	C2BB	ED5B07C2	ld	de,(xe)	; if xs-xe=0 then return
89	C2BF	B7	or	a	
90	C2C0	ED52	sbc	hl,de	
91	C2C2	282F	jr	z,lexit	
92	C2C4	2A03C2	ld	hl,(xs)	; xs:=xs+sx
93	C2C7	ED5B11C2	ld	de,(sx)	
94	C2CB	19	add	hl,de	
95	C2CC	2203C2	ld	(xs),hl	
96	C2CF	2A0FC2	ld	hl,(ee)	; ee:=ee+dy
97	C2D2	ED5B0DC2	ld	de,(dy)	
98	C2D6	19	add	hl,de	
99	C2D7	220FC2	ld	(ee),hl	
100	C2DA	ED5B0BC2	ld	de,(dx)	; if ee>=dx then
101	C2DE	B7	or	a	
102	C2DF	ED52	sbc	hl,de	
103	C2E1	38BD	jr	c,l5	; goto 15
104	C2E3	220FC2	ld	(ee),hl	; ee:=ee-dx
105	C2E6	2A05C2	ld	hl,(ys)	; ys:=ys+sy
106	C2E9	ED5B13C2	ld	de,(sy)	
107	C2ED	19	add	hl,de	
108	C2EE	2205C2	ld	(ys),hl	
109	C2F1	18AD	jr	l5	; goto 15
110	C2F3	C9	lexit: ret		
111					
112	C2F4	7C	abs: ld	a,h	; hl:=abs(hl)
113	C2F5	FE80	cp	80H	
114	C2F7	3806	jr	c,absex	
115	C2F9	2F	cpl		
116	C2FA	67	ld	h,a	
117	C2FB	7D	ld	a,l	
118	C2FC	2F	cpl		
119	C2FD	6F	ld	l,a	
120	C2FE	23	inc	hl	
121	C2FF	B7	absex: or	a	; carry:=0
122	C300	C9	ret		



## 2-2-2 CIRCLE

円を描くにもいろいろな方法があります。実数演算を使っても良いなら円の方程式,

$$x^2 + y^2 = r^2$$

から,

$$y = \pm \sqrt{r^2 - x^2}$$

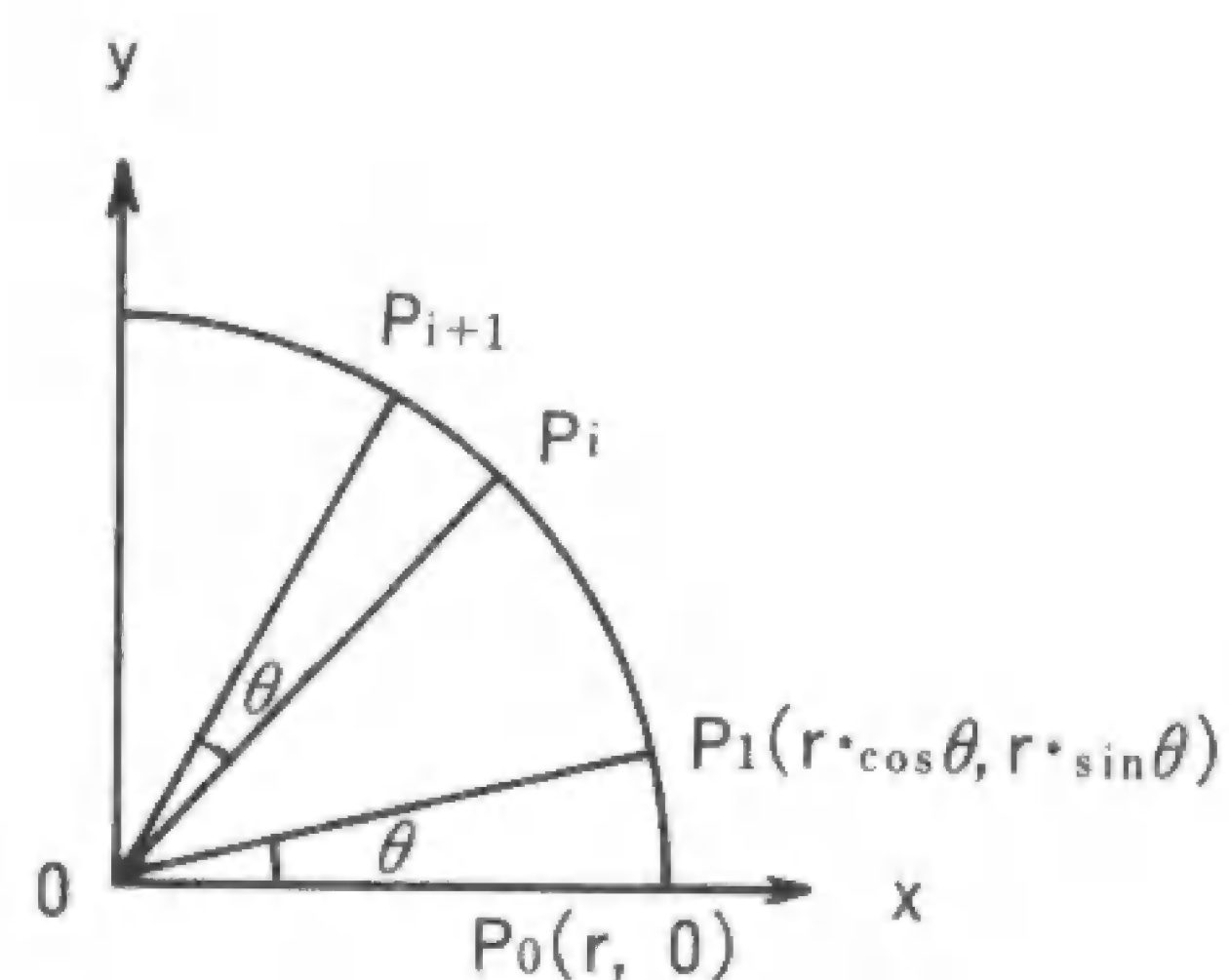
として,  $x$  の値を  $r$  から  $-r$  まで変化させながら求める方法があります。また, 媒介変数  $\theta$  を使って円周上の点を回転させながら描いていく方法もあります。この方法で円を描く場合, 原点を中心とした点  $P_i$  を左回りに  $\theta$  だけ回転させた点  $P_{i+1}$  は

$$x_{i+1} = x_i \cdot \cos \theta - y_i \cdot \sin \theta$$

$$y_{i+1} = x_i \cdot \sin \theta + y_i \cdot \cos \theta$$

として求められます (図 2-5)。

図2-5 円の座標値



ここで  $\theta$  が微小ならば,

$$\sin \theta \doteq \theta, \cos \theta \doteq 1$$

となり, これより,



$$\begin{aligned}x_{i+1} &= x_i - \theta y_i \\ y_{i+1} &= \theta x_i + y_i\end{aligned}$$

となります。ところがこの近似は意外にあらっぽくこの式から円を描こうとすると、半径がだんだん大きくなり、うずまき状になってしまい円にはなりません。そこでこの誤差を補正するために、第2式の  $x_i$  を  $x_{i+1}$  に置き換えます。結局、式は次のようになります。

$$\begin{aligned}x_{i+1} &= x_i - \theta y_i \\ y_{i+1} &= \theta x_{i+1} + y_i\end{aligned}$$

これを1周するまで繰り返せばよいことになります。ここで  $\theta$  の値の決め方ですが、 $2^{-n}$  とすればマシン語でも乗算をせずに、シフト演算だけでできます。

リスト2-6はこの方法で円を描いたものです。 $\theta$  の値（リスト中では変数E）を  $2^{-8}$  ( $\frac{1}{256}$ ) にしました。256で割るというのはマシン語では簡単にかけます（符号なしの場合）。たとえば、deレジスタの値を256で割るプログラムは次のようになります（小数部は無視）。

```
ld  e, d
ld  d, 0
```

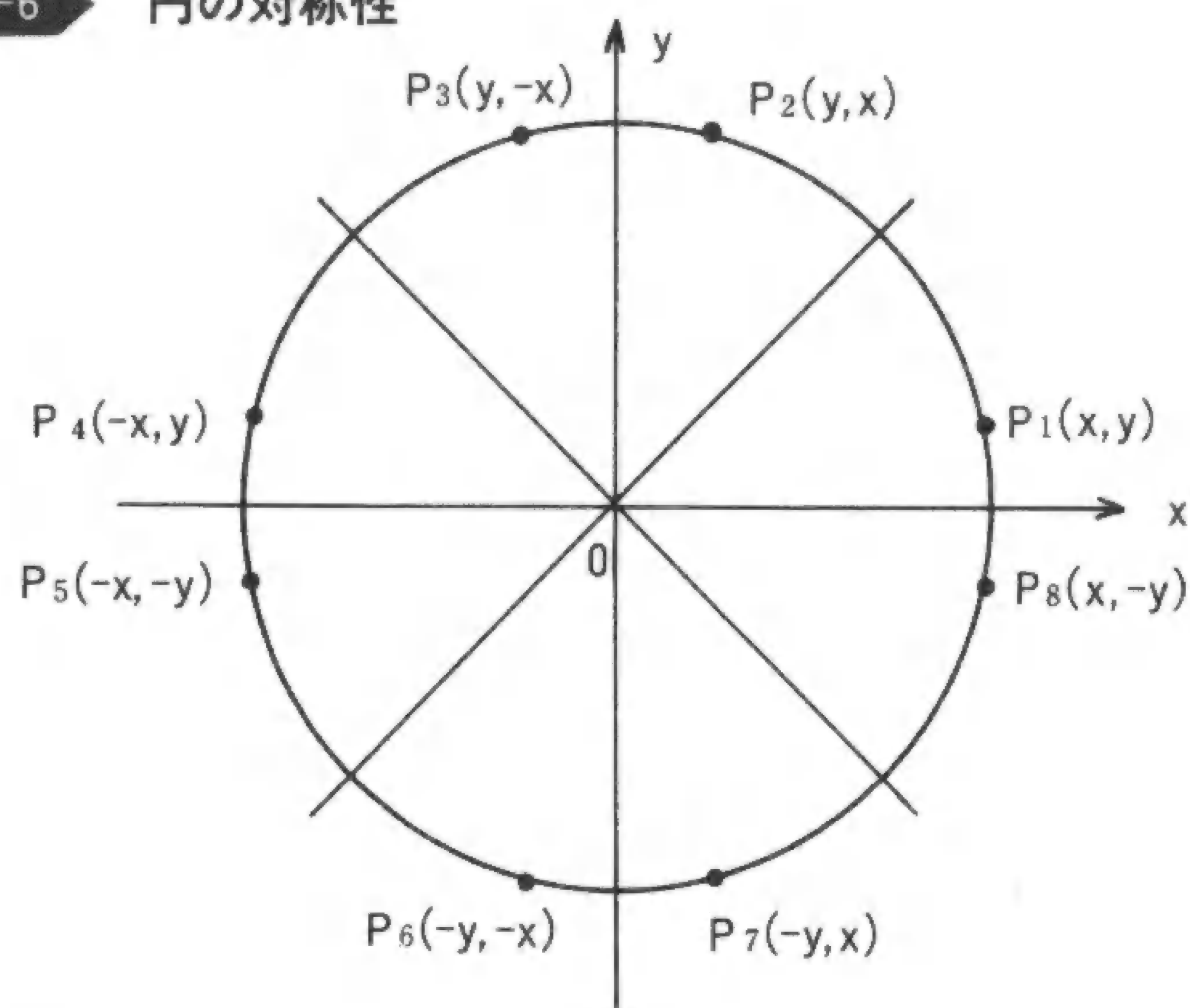
このプログラムでは、座標原点が（0，0）ではなく、（x 0，y 0）なので1200行で座標変換をしています。また、X 1の640 × 200ドット・モードでは画面の縦横比（アスペクト比）が約X：Y＝1：2なので、1200行ではこのことも考慮に入れています。1070行ではBASICのCIRCLE命令の半径がY方向を基準にしているため半径Rを2倍にしています。1110行のループ回数は経験値です。このプログラムを実行した後、BASICのCIRCLE命令を実行してみるとほぼ重なることがわかると思います。

このプログラムをこのままマシン語化してもいいのですが、もう一步高速化のために工夫してみましょう。PC-8801



や MSX などでは CIRCLE 命令を実行すると、同時に 8 ドットずつ点を打っています。これは図 2-6 のように、円の対称性を利用したものです。また半径の小さな円の場合、式で求めた  $x_i$  と  $x_{i+1}$  ,  $y_i$  と  $y_{i+1}$  が同じ値になってしまふことが多くなります。そこで前に求めた座標値と新しく求めた座標値が一致していれば何もしない、というようにします。

図2-6 円の対称性



リスト2-6

```

1000 INIT : WIDTH 80 : CLS 4
1010 :
1020 INPUT "X0,Y0";X0,Y0
1030 INPUT "RADIUS";R
1040 INPUT "COLOR";C
1050 :
1060 E=1/256
1070 X=R+R
1080 Y=0
1090 LOOP=0
1100 :
1110 WHILE LOOP < 1616
1120     GOSUB "pset"
1130     X=X-E*Y
1140     Y=E*X+Y
1150     LOOP=LOOP+1
1160 WEND
1170 END
1180 :
1190 LABEL "pset"
1200     XT=X0+X : YT=Y0+Y/2
1210     PSET (XT,YT,C)
1220 RETURN

```



以上の2点を加味したものが、リスト2-7です。これをマシン語化したのがリスト2-8です。ここで注意してほしいのが、座標値を計算する式は整数演算ではなく、固定小数点演算であるということです。もともと乗算部分は小数点以下の部分であり、これがオーバーフローしたときのみ（キャリーフラグが立ったときのみ）'intx'、'inty'の値が変わっていくからです。

## リスト2-7

```

1000 INIT : WIDTH 80 : CLS 4
1010 :
1020 INPUT "X0,Y0";X0,Y0
1030 INPUT "RADIUS";R
1040 INPUT "COLOR";C
1050 :
1060 E=1/256
1070 X=R+R : Y=0
1080 XB=X : YB=Y
1090 LOOP=0
1100 :
1110 GOSUB "pset"
1120 WHILE LOOP < 202
1130     X=X-E*Y
1140     Y=E*X+Y
1150     LOOP=LOOP+1
1160     GOSUB "compare"
1170     IF FLAG=1 THEN 1190
1180     GOSUB "pset"
1190 WEND
1200 END
1210 :
1220 LABEL "compare"
1230     IF INT(X)=INT(XB) AND INT(Y)=INT(YB)
        THEN FLAG=1 : RETURN
1240     XB=X : YB=Y
1250     FLAG=0
1260 RETURN
1270 :
1280 LABEL "pset"
1290     XT=X0+X : YT=Y0+Y/2
1300     PSET (XT,YT,C)
1310     XT=X0+X : YT=Y0-Y/2
1320     PSET (XT,YT,C)
1330     XT=X0+Y : YT=Y0+X/2
1340     PSET (XT,YT,C)
1350     XT=X0+Y : YT=Y0-X/2
1360     PSET (XT,YT,C)
1370     XT=X0-X : YT=Y0+Y/2
1380     PSET (XT,YT,C)
1390     XT=X0-X : YT=Y0-Y/2
1400     PSET (XT,YT,C)
1410     XT=X0-Y : YT=Y0+X/2
1420     PSET (XT,YT,C)
1430     XT=X0-Y : YT=Y0-X/2
1440     PSET (XT,YT,C)
1450 RETURN

```



リスト2-8

```

1      :
2      :      --- subroutine 'circle' ---
3      :
4      :      entry      (xo)      := center point x
5      :                  (yo)      := center point y
6      :                  (rad)     := radius
7      :                  (color)   := dot color
8      :
9      :      modifies   af, bc, de, hl
10     :
11     11 C110      pset2 equ      0C110H
12     12 C107      color equ     0C107H
13     13           ;
14     14 C300      org      0C300H
15     15 C300 C313C3      jp      circle
16     16           ;
17     17 C303 4001      xo:      defw 320
18     18 C305 6400      yo:      defw 100
19     19 C307 4B00      rad:     defw 75
20     20           ;
21     21 C309      intx:      defs 2
22     22 C30B      fracx:     defs 1
23     23 C30C      workx:     defs 2
24     24 C30E      inty:      defs 2
25     25 C310      fracy:     defs 1
26     26 C311      worky:     defs 2
27     27           ;
28     28 C313      circle equ  $
29     29           ;
30     30 C313 2109C3      ld      hl, intx      ; work <- zero clear
31     31 C316 AF          xor      a
32     32 C317 060A      ld      b, 10
33     33 C319 77          clrwrk: ld      (hl), a
34     34 C31A 23          inc      hl
35     35 C31B 10FC      djnz     clrwrk
36     36           ;
37     37 C31D CD7FC3      call     circl
38     38 C320 06CC      ld      b, 204
39     39 C322 C5          cloop:  push   bc
40     40 C323 2A09C3      ld      hl, (intx)
41     41 C326 ED5B0EC3    ld      de, (inty)
42     42 C32A 3A10C3      ld      a, (fracy)
43     43 C32D 43          ld      b, e
44     44 C32E 5A          ld      e, d
45     45 C32F 1600      ld      d, 0      ; y*e
46     46 C331 80          add      a, b
47     47 C332 3210C3      ld      (fracy), a
48     48 C335 ED52      sbc      hl, de
49     49 C337 2209C3      ld      (intx), hl
50     50 C33A 3A0BC3      ld      a, (fracx)
51     51 C33D 45          ld      b, 1
52     52 C33E 6C          ld      l, h
53     53 C33F 2600      ld      h, 0      ; x*e
54     54 C341 80          add      a, b
55     55 C342 320BC3      ld      (fracx), a
56     56 C345 ED5B0EC3    ld      de, (inty)
57     57 C349 ED5A      adc      hl, de
58     58 C34B 220EC3      ld      (inty), hl
59     59 C34E CD5AC3      call     longcp
60     60 C351 3003      jr      nc, next
61     61 C353 CD7FC3      call     circl
62     62 C356 C1          next:   pop     bc
63     63 C357 10C9      djnz     cloop
64     64 C359 C9          ret
65     65           ;
66     66           ; if (worky=inty) and (workx=intx) then cy flag off
67     67           ; else cy flag on
68     68 C35A ED5B11C3    longcp: ld      de, (worky)
69     69 C35E B7          or      a
70     70 C35F ED52      sbc      hl, de      ; inty = worky ?
71     71 C361 200C      jr      nz, noteql      ; no, then goto "exit"
72     72 C363 2A09C3      ld      hl, (intx)
73     73 C366 ED5B0CC3    ld      de, (workx)

```



## リスト2-8 つづき

```

74 C36A B7          or      a
75 C36B ED52        sbc     hl,de      ; intx = workx ?
76 C36D 280E        jr      z,equal    ;      yes,then goto "equal"
77 C36F 2A09C3      noteql: ld     hl,(intx)
78 C372 220CC3      ld      (workx),hl ; workx:=intx
79 C375 2A0EC3      ld      hl,(inty)
80 C378 2211C3      ld      (worky),hl ; worky:=inty
81 C37B 37          scf
82 C37C C9          ret
83 C37D B7          equal:  or      a      ; cy flag off
84 C37E C9          ret
85
86      ; ---  circ1  ---
87
88      ; xt:=xo+x , yt:=yo+y/2
89 C37F 2A03C3      circ1:  ld      hl,(xo)
90 C382 ED5B09C3    ld      de,(intx)
91 C386 19          add     hl,de
92 C387 44          ld      b,h
93 C388 4D          ld      c,l
94 C389 2A05C3      ld      hl,(yo)
95 C38C ED5B0EC3    ld      de,(inty)
96 C390 CB3A        srl     d
97 C392 CB1B        rr      e
98 C394 19          add     hl,de
99 C395 EB          ex      de,hl
100 C396 CD10C1      call    pset2
101      ; xt:=xo+x , yt:=yo-y/2
102 C399 2A03C3      ld      hl,(xo)
103 C39C ED5B09C3    ld      de,(intx)
104 C3A0 19          add     hl,de
105 C3A1 44          ld      b,h
106 C3A2 4D          ld      c,l
107 C3A3 2A05C3      ld      hl,(yo)
108 C3A6 ED5B0EC3    ld      de,(inty)
109 C3AA CB3A        srl     d
110 C3AC CB1B        rr      e
111 C3AE B7          or      a
112 C3AF ED52        sbc     hl,de
113 C3B1 EB          ex      de,hl
114 C3B2 CD10C1      call    pset2
115      ; xt:=xo+y , yt:=yo+x/2
116 C3B5 2A0EC3      ld      hl,(inty)
117 C3B8 ED5B03C3    ld      de,(xo)
118 C3BC 19          add     hl,de
119 C3BD 44          ld      b,h
120 C3BE 4D          ld      c,l
121 C3BF 2A05C3      ld      hl,(yo)
122 C3C2 ED5B09C3    ld      de,(intx)
123 C3C6 CB3A        srl     d
124 C3C8 CB1B        rr      e
125 C3CA 19          add     hl,de
126 C3CB EB          ex      de,hl
127 C3CC CD10C1      call    pset2
128      ; xt:=xo+y , yt:=yo-x/2
129 C3CF 2A0EC3      ld      hl,(inty)
130 C3D2 ED5B03C3    ld      de,(xo)
131 C3D6 19          add     hl,de
132 C3D7 44          ld      b,h
133 C3D8 4D          ld      c,l
134 C3D9 2A05C3      ld      hl,(yo)
135 C3DC ED5B09C3    ld      de,(intx)
136 C3E0 CB3A        srl     d
137 C3E2 CB1B        rr      e
138 C3E4 B7          or      a
139 C3E5 ED52        sbc     hl,de
140 C3E7 EB          ex      de,hl
141 C3E8 CD10C1      call    pset2
142      ; xt:=xo-x , yt:=yo+y/2
143 C3EB 2A03C3      ld      hl,(xo)
144 C3EE ED5B09C3    ld      de,(intx)
145 C3F2 B7          or      a
146 C3F3 ED52        sbc     hl,de
147 C3F5 44          ld      b,h

```

つづく



リスト2-8 つづき

```

148 C3F6 4D          ld      c,l
149 C3F7 2A05C3      ld      hl,(yo)
150 C3FA ED5B0EC3    ld      de,(inty)
151 C3FE CB3A        srl     d
152 C400 CB1B        rr      e
153 C402 19          add     hl,de
154 C403 EB          ex      de,hl
155 C404 CD10C1      call    pset2
156                  ; xt:=xo-x , yt:=yo-y/2
157 C407 2A03C3      ld      hl,(xo)
158 C40A ED5B09C3    ld      de,(intx)
159 C40E B7          or      a
160 C40F ED52        sbc     hl,de
161 C411 44          ld      b,h
162 C412 4D          ld      c,l
163 C413 2A05C3      ld      hl,(yo)
164 C416 ED5B0EC3    ld      de,(inty)
165 C41A CB3A        srl     d
166 C41C CB1B        rr      e
167 C41E B7          or      a
168 C41F ED52        sbc     hl,de
169 C421 EB          ex      de,hl
170 C422 CD10C1      call    pset2
171                  ; xt:=xo-y , yt:=yo+x/2
172 C425 2A03C3      ld      hl,(xo)
173 C428 ED5B0EC3    ld      de,(inty)
174 C42C B7          or      a
175 C42D ED52        sbc     hl,de
176 C42F 44          ld      b,h
177 C430 4D          ld      c,l
178 C431 2A05C3      ld      hl,(yo)
179 C434 ED5B09C3    ld      de,(intx)
180 C438 CB3A        srl     d
181 C43A CB1B        rr      e
182 C43C B7          or      a
183 C43D ED52        sbc     hl,de
184 C43F EB          ex      de,hl
185 C440 CD10C1      call    pset2
186                  ; xt:=xo-y , yt:=yo-x/2
187 C443 2A03C3      ld      hl,(xo)
188 C446 ED5B0EC3    ld      de,(inty)
189 C44A B7          or      a
190 C44B ED52        sbc     hl,de
191 C44D 44          ld      b,h
192 C44E 4D          ld      c,l
193 C44F 2A05C3      ld      hl,(yo)
194 C452 ED5B09C3    ld      de,(intx)
195 C456 CB3A        srl     d
196 C458 CB1B        rr      e
197 C45A 19          add     hl,de
198 C45B EB          ex      de,hl
199 C45C C310C1      jp      pset2
200                  ;

```

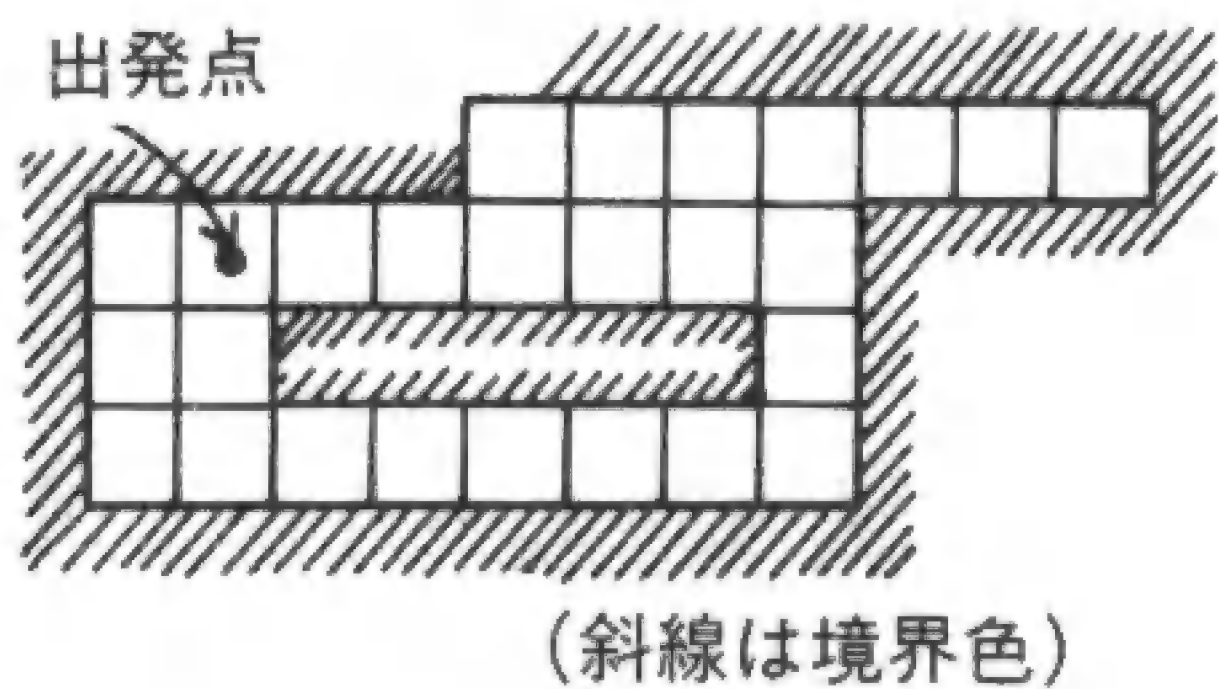


# 2-3 PAINT

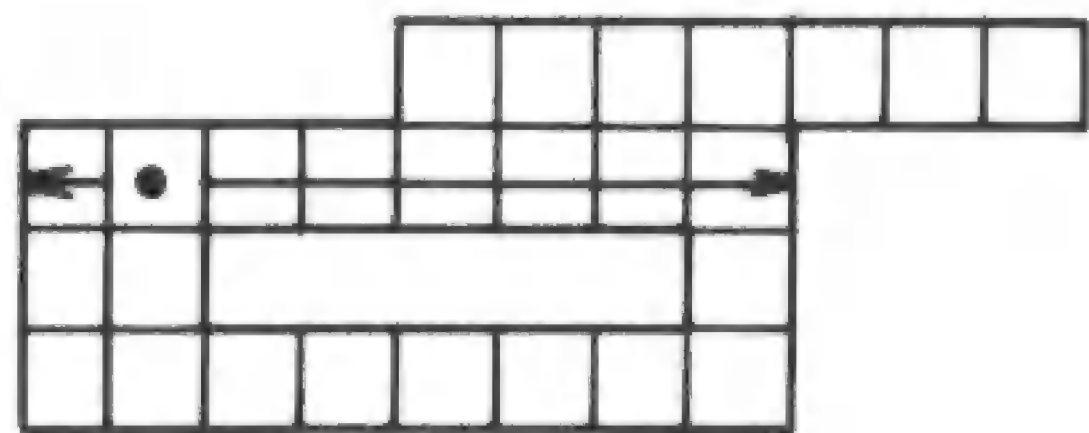
ここで紹介する PAINT ルーチンは、もっとも一般的なベーシック・フィルアルゴリズムを使ったものです。

まず、始めに単色の PAINT について考えてみます。いま、図 2-7 のような領域を塗りつぶすとする処理の順序は次のようになります（図の 1 マスは 1 ドットに相当するものとする）。

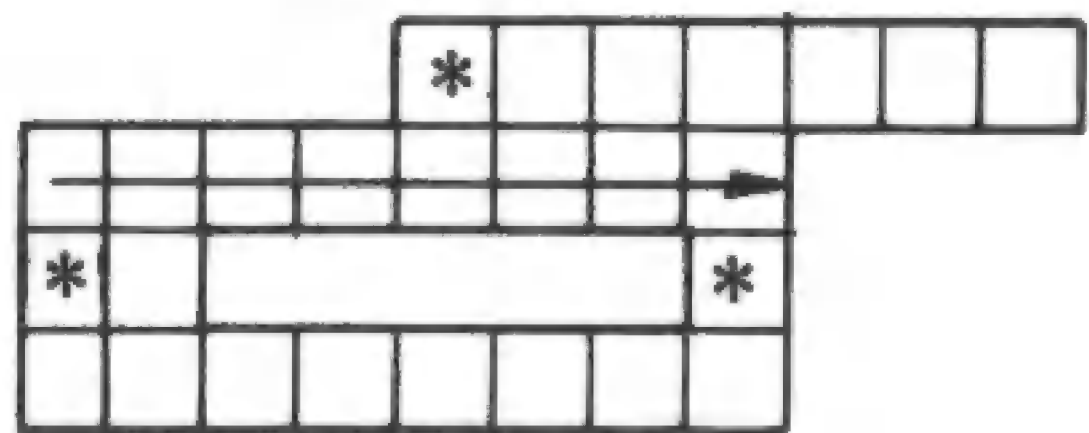
図2-7 PAINTの処理



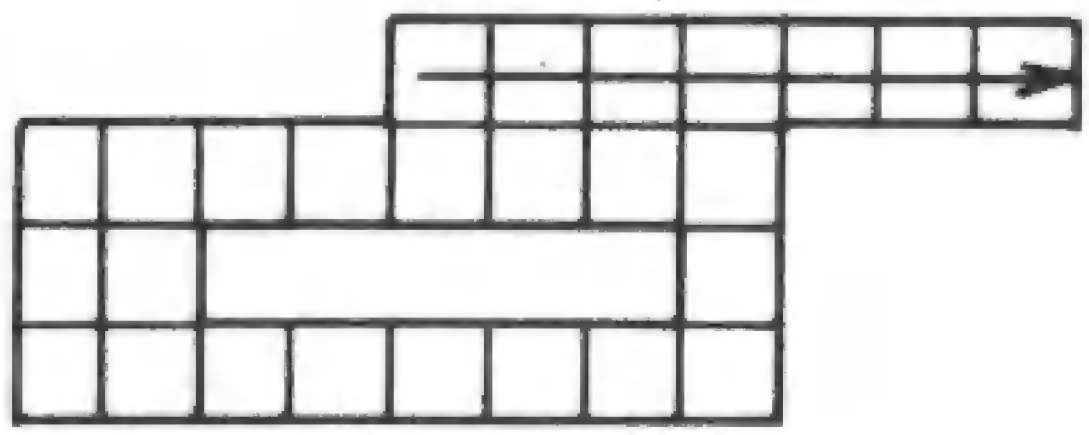
● 出発点をスタックに積む



● 左、右をサーチして塗る



● 矢印の上下を見て\*印をスタックに積む



● 最後に積んだものをポップして同様の処理を繰り返す

- ① 出発点をスタックに積む。
- ② 境界色になるか、画面の左端（X座標が 0）になるまで左へ色をサーチする。この結果を X 1 とする。
- ③ 境界色になるか、画面の右端（X座標が 639）になるまで



右へ色をサーチする。この結果をX 2 とする。

④ X 1 から X 2 まで点を打つ (LINE で代用できる)。

⑤ X 1 から X 2 までの上下の点を見て、もし境界色でないならその座標をスタックに積む。このとき隣合う点をスタックに積まないようにフラグを設ける。

⑥ スタックが空になるまで②から繰り返す。

さて、⑤のとき、すでに塗ってあるかどうかということ判断しなければなりません。単色 PAINT の場合、上下の点が塗る色であれば、すでに塗ったものとして判断できます。このアルゴリズムをもとにつくったプログラムがリスト 2-9 です。1280 行から 1430 行が上下の点を調べるものですが、境界色だけでなく塗る色とも条件判断していることに注目してください。つまり、ここでは境界色と塗る色を同等に扱っていることになります。

ここで変数 LF, UF は隣合う点をスタックに積まないようにするフラグ、CUF, CLF は Y 座標の範囲を越えないようにするためのフラグです。スタックは配列で代用しました。リスト 2-10 はテスト用のプログラムです。

中間色 PAINT の場合、単色 PAINT と違って問題になるのは、どうやってすでに塗ったかを判断することです。これは上下のドットの色からは単純に判断できないからです。この方法にはいろいろあるようですが、ここでは塗るときに境界色と境界色の間を見てすでに塗られているかどうか判断するようにしました。リスト 2-11 の 1790 行以降のサブルーチンがその判断をするものです。

どのライン・パターンで塗るかは、Y 座標の値で決めることにします (1450 行)。配列変数 TILECOLOR は与えられたライン・パターンのデータからドット単位の色を格納するものです。1230 行から 1260 行のデータをいろいろ変更して試してみてください。

このプログラムをマシン語化するには、まず POINT ルーチンをつくらなければなりません。しかし、POINT ルーチンはすでにみてきた PSET ルーチンの逆の動作をするも



のですから簡単につくることができます。プログラムをリスト 2-12 に示しますが、もはや説明の必要はないと思います。

## リスト 2-9

```

1000 DEFINT A-Z
1010 DIM SX(100),SY(100)
1020 :
1030 Fon=1 : Foff=0
1040 :
1050 INPUT "X0,Y0";X0,Y0
1060 INPUT "PAINT COLOR";PAC
1070 INPUT "BORDER COLOR";BDC
1080 :
1090 I=0
1100 SX(I)=X0 : SY(I)=Y0 : I=1
1110 LABEL "MAIN"
1120     I=I-1
1130     IF I<0 THEN END
1140     X=SX(I) : Y=SY(I)
1150 :
1160     WHILE POINT(X,Y)<>BDC AND X>=0
1170         X=X-1
1180     WEND
1190     X1=X+1
1200     X=SX(I)
1210     WHILE POINT(X,Y)<>BDC AND X<=639
1220         X=X+1
1230     WEND
1240     X2=X-1
1250 :
1260     LINE (X1,Y)-(X2,Y),PSET,PAC
1270 :
1280     CUF=Fon : CLF=Fon
1290     IF Y=0 THEN CUF=Foff
1300     IF Y=199 THEN CLF=Foff
1310     UF=Foff : LF=Foff
1320     FOR X=X1 TO X2
1330         IF CLF=Foff THEN "SKIP1"
1340             PTL=POINT(X,Y+1)
1350             IF PTL<>BDC AND PTL<>PAC AND LF=Foff
                THEN LF=Fon : SX(I)=X : SY(I)=Y+1 : I=I+1
1360             IF PTL=BDC OR PTL=PAC AND LF=Fon
                THEN LF=Foff
1370         LABEL "SKIP1"
1380         IF CUF=Foff THEN "SKIP2"
1390             PTU=POINT(X,Y-1)
1400             IF PTU<>BDC AND PTU<>PAC AND UF=Foff
                THEN UF=Fon : SX(I)=X : SY(I)=Y-1 : I=I+1
1410             IF PTU=BDC OR PTU=PAC AND UF=Fon
                THEN UF=Foff
1420         LABEL "SKIP2"
1430     NEXT
1440 GOTO "MAIN"

```

## リスト 2-10

```

10 INIT : WIDTH 80 : CLS 4
20 LINE (100,100)-(120,100)-(120,120)-(130,120)-(130,100)
30 LINE -(150,100)-(150,180)-(100,180)-(100,100)-(100,150)
40 LINE -(90,150)-(90,160)-(70,160)-(70,140)-(100,140)-(100,100)
50 LINE (115,140)-(135,140)-(135,160)-(115,160)-(115,140)

```



```

1000 DEFINT A-Z
1010 DEF FNCH(A$,J)=VAL(MID$(RIGHT$("00000000"+BIN$(VAL("&h"+A$)),8),J+1,1))
1020 DIM SX(100),SY(100)
1030 :
1040 Fon=-1 : Foff=0
1050 :
1060 RESTORE "TILE"
1070 READ No
1080 DIM TILE$(No-1),TILECOLOR(No-1,7)
1090 FOR I=0 TO No*3-1 STEP 3
1100     A$=""
1110     FOR J=0 TO 2
1120         READ A$ : TILE$(I/3)=TILE$(I/3)+A$
1130     NEXT J,I
1140 RESTORE "TILE"
1150 READ TMP          : ' DAMMT READ
1160 FOR I=0 TO No*3-1 STEP 3
1170     READ A$ : READ B$ : READ C$
1180     FOR J=0 TO 7
1190         TILECOLOR(I/3,J)=FNCH(A$,J)+FNCH(B$,J)*2+FNCH(C$,J)*4
1200     NEXT J,I
1210 :
1220 LABEL "TILE"
1230 DATA 3          : ' TILE DATA No.
1240 DATA AA,CC,AA   : ' TILE DATA (B,R,G)
1250 DATA 55,11,22
1260 DATA 22,33,44
1270 :
1280 INPUT "X0,Y0";X0,Y0
1290 INPUT "BORDER COLOR";BDC
1300 :
1310 I=0
1320 SY(I)=Y0 : SX(I)=X0
1330 I=1
1340 LABEL "MAIN"
1350     I=I-1
1360     IF I<0 THEN END
1370     X=SX(I) : Y=SY(I)
1380     :
1390     GOSUB "LEFT_SEARCH" : X=SX(I)
1400     GOSUB "RIGHT_SEARCH"
1410     :
1420     GOSUB "CHECK"
1430     IF FLAG THEN "MAIN"
1440     :
1450     LINE (X1,Y)-(X2,Y),PSET,BF,HEXCHR$(TILE$(Y MOD No))
1460     :
1470     CUF=Fon : CLF=Fon
1480     IF Y=0 THEN CUF=Foff
1490     IF Y=199 THEN CLF=Foff
1500     UF=Foff : LF=Foff
1510     FOR X=X1 TO X2
1520         IF CLF=Foff THEN "SKIP1"
1530             PTL=POINT(X,Y+1)
1540             IF PTL<>BDC AND LF=Foff
1550                 THEN LF=Fon : SX(I)=X : SY(I)=Y+1 : I=I+1
1560             IF PTL=BDC AND LF=Fon THEN LF=Foff
1570             LABEL "SKIP1"
1580             IF CUF=Foff THEN "SKIP2"
1590                 PTU=POINT(X,Y-1)
1600                 IF PTU<>BDC AND UF=Foff
1610                     THEN UF=Fon : SX(I)=X : SY(I)=Y-1 : I=I+1
1620                 IF PTU=BDC AND UF=Fon THEN UF=Foff
1630             LABEL "SKIP2"
1640     NEXT
1650 GOTO "MAIN"
1660 :
1670 LABEL "LEFT_SEARCH"
1680     WHILE POINT(X,Y)<>BDC AND X>=0
1690         X=X-1
1700     WEND
1710     X1=X+1
1720 RETURN
1730 :

```



## リスト2-11 つづき

```

1720 LABEL "RIGHT_SEARCH"
1730     WHILE POINT(X,Y)<>BDC AND X<=639
1740         X=X+1
1750     WEND
1760     X2=X-1
1770 RETURN
1780 :
1790 LABEL "CHECK"
1800     FLAG=Fon
1810     X=X1
1820     LABEL "LOOP"
1830         IF POINT(X,Y)<>TILECOLOR(Y MOD No , X MOD 8)
            THEN FLAG=Foff : RETURN
1840         X=X+1
1850         IF X<=X2 THEN "LOOP"
1860 RETURN

```

## リスト2-12

```

1      :
2      :      ---      subroutine 'point' ---
3      :      entry      (x0)      := point x
4      :      :      (y0)      := point y
5      :      :      returns      (ix) or (pcolor)
6      :      :      modifies      af, bc, de, hl
7      :
8      C000      calcva equ      0C000H
9      :
10     C500      :      org      0C500H
11     C500 C307C5      :      jp      point
12     :
13     C503 0000      x0:      defw      0000H
14     C505 0000      y0:      defw      0000H
15     :
16     C507      point equ      $
17     :
18     C507 ED4B03C5      point1: ld      bc, (x0)
19     C50B ED5B05C5      :      ld      de, (y0)
20     C50F 217F02      point2: ld      hl, 639
21     C512 B7      :      or      a
22     C513 ED42      :      sbc      hl, bc
23     C515 D8      :      ret      c
24     C516 21C700      :      ld      hl, 199
25     C519 ED52      :      sbc      hl, de
26     C51B D8      :      ret      c
27     :
28     C51C DD216EC5      point3: ld      ix, pcolor
29     C520 DD360000      :      ld      (ix), 0      ; color:=0
30     C524 79      :      ld      a, c
31     C525 2F      :      cpl
32     C526 E607      :      and      7      ; a:=x0 mod 8
33     C528 326DC5      :      ld      (xmod8), a
34     C52B CD00C0      :      call      calcva
35     C52E 3A6DC5      :      ld      a, (xmod8)
36     C531 87      :      add      a, a      ; shift left 3 bits
37     C532 87      :      add      a, a
38     C533 87      :      add      a, a
39     C534 F643      :      or      43H      ; or 01000011b
40     C536 3248C5      :      ld      (@bitb), a
41     C539 3257C5      :      ld      (@bitr), a
42     C53C 3266C5      :      ld      (@bitg), a
43     :
44     C53F 110040      tstblu: ld      de, 4000H      ; blue
45     C542 19      :      add      hl, de
46     C543 44      :      ld      b, h
47     C544 4D      :      ld      c, l
48     C545 ED58      :      in      e, (c)
49     C547 CB      :      defb      0CBH      ; bit n,e
50     C548      @bitb:      defs      1
51     C549 2804      :      jr      z, tstred
52     C54B DDCB00C6      :      set      0, (ix)

```

つづく



リスト2-12 つづき

```

53 C54F 110040      tstred: ld      de,4000H      ; red
54 C552 19          add      hl,de
55 C553 44          ld       b,h
56 C554 ED58        in       e,(c)
57 C556 CB          defb     0CBH      ; bit n,e
58 C557             @bitr: defs     1
59 C558 2804         jr       z,tstgrn
60 C55A DDCB00CE     set      1,(ix)
61 C55E 110040      tstgrn: ld      de,4000H      ; green
62 C561 19          add      hl,de
63 C562 44          ld       b,h
64 C563 ED58        in       e,(c)
65 C565 CB          defb     0CBH      ; bit n,e
66 C566             @bitg: defs     1
67 C567 C8          ret       z
68 C568 DDCB00D6     set      2,(ix)
69 C56C C9          ret
70                  ;
71 C56D             xmod8: defs     1
72 C56E             pcolor: defs    1

```

リスト 2-13 が中間色 PAINT ルーチンです。リスト 2-11 をほぼそのままハンド・コンパイルしたものですから、特に解説する部分はありません。ただ、‘save 1’や‘save 2’のようにスタックを操作するときは変なデータをプッシュしないように注意しなければなりません。また、スタックがオーバーフローしないように監視することも必要です。なお、ライン・パターンは 8 個までとしています。‘tilec’のエリアを大きくすれば数を増やすことができます。この PAINT ルーチンでは、座標値のチェックをする必要がないので、‘pset 3’、‘point 3’をコールしています。

ここでは、スタックからポップした後、塗られているかどうかチェックしていますが、これをスタックに積むときにチェックしてもよいでしょう。こうすると、まず左右の境界色を調べるわけですからスタックに積むのは X 座標の左右端と Y 座標となります。そして、スタックからポップしたらすぐ塗ることにします。

さて、リスト 2-13 の実行速度ですが、最初の X 1 の BASIC (CZ-8 FB 01) よりは速いのですが、X 1 Turbo や“Oh!MZ 誌”に掲載された PAINT よりは遅くなってしまいました。

プログラムの実行速度をあげたいときは、もっとも数多く通るルーチンを高速化すればよいので、ここでは POINT



と PSET ルーチンを高速化すればよいことがわかると思います。ここでは境界色をサーチするときも、点を打つときも常に Y 座標は一定です。ですから、毎回 'calcva' という VRAM アドレスを求めるルーチンを使わずに、始めに 1 回だけこれをコールして、あとは求まった VRAM アドレスをインクリメント、デクリメントしてカラーコードを得るなり、点を打つなりすればかなり速くなると思います。

こうすると、VRAM アドレスと座標値を両方管理しなければならないので、プログラムは複雑に、なおかつ長くなるでしょう。また、この場合、POINT や PSET ルーチンも汎用のものでなく、PAINT 専用のものになります。もっと速くしたい方はチャレンジしてみてください。

ちょっと余談になりますが、“Oh!MZ 誌”に掲載された PAINT ルーチンでは、VRAM の表示されないエリア (7FD0~7FFF, BFD0~BFFF, FFD0~FFFF) に一旦ほかのプログラムを待避させて、その空いた領域をスタック・エリアにするという方法をとっていました。たいへんうまい方法だと思います。

また、スタックというのは LIFO (Last IN First Out) バッファですが、X1 Turbo では FIFO (First IN First Out) バッファを使って PAINT しています。つまり、最初に積んだものを最初に塗るということです (塗る順序に注目してください)。確かに FIFO バッファを使うとワーク・エリアのサイズを小さくできますが、その代わりポインタを余計に持たなければならないので、その分処理がめんどうになります。



```

1      ;
2      ; --- subroutine 'tiling paint' ---
3      ;
4      ;       entry   (x0)   := origin x
5      ;               (y0)   := origin y
6      ;               (bdc)  := border color
7      ;               (no)   := tile data no.
8      ;               tile   := tile data (b,r,g)
9      ;       modify  af, bc, de, hl, ix, iy
10     C51C      point3 equ    0C51CH
11     C11D      pset3  equ    0C11DH
12     000B      print  equ    000BH
13     ;
14     CF00      stktop equ    0CF00H
15     C800      stkbot equ    0C800H
16     C107      color  equ    0C107H
17     C56E      pcolor equ    0C56EH
18     ;
19     D000      ;
20     D000 C352D0      org    0D000H
21     ;               jp      hpaint
22     D003 6400      x0:    defw 100
23     D005 5000      y0:    defw 80
24     D007 04        bdc:   defb 4
25     D008 03        no:    defb 3          ; max 8
26     D009 AACCAA      tile: defb 0AAH, 0CCH, 0AAH
27     D00C 551122      defb 55H, 11H, 22H
28     D00F 223344      defb 22H, 33H, 44H
29     D012      tilec: defs 64
30     ;
31     ;
32     D052      hpaint equ    $
33     ;
34     D052 ED4B03D0      ld      bc, (x0)          ; check origin point
35     D056 ED5B05D0      ld      de, (y0)
36     D05A 217F02      ld      hl, 639
37     D05D B7          or      a
38     D05E ED42      sbc      hl, bc
39     D060 D8          ret      c
40     D061 21C700      ld      hl, 199
41     D064 ED52      sbc      hl, de
42     D066 D8          ret      c
43     ;
44     D067 E1          pop      hl
45     D068 228ED1      ld      (retadr), hl      ; save return address
46     ;
47     ; --- make color table ---
48     ;
49     D06B 2112D0      ctbl:   ld      hl, tilec
50     D06E 1113D0      ld      de, tilec+1
51     D071 013F00      ld      bc, 63
52     D074 3600      ld      (hl), 0
53     D076 EDB0      ldir          ; tilec[] := 0
54     ;
55     D078 3A08D0      ld      a, (no)
56     D07B 47          ld      b, a
57     D07C DD2109D0      ld      ix, tile
58     D080 FD2112D0      ld      iy, tilec
59     D084 DD5600      ctbl1: ld      d, (ix)          ; d <- blue
60     D087 DD5E01      ld      e, (ix+1)          ; e <- red
61     D08A DD6E02      ld      l, (ix+2)          ; l <- green
62     D08D 2608      ld      h, 8          ; loop counter
63     D08F CB22      ctbl2: sla      d
64     D091 3004      jr      nc, ctbl3
65     D093 FDCB00C6      set      0, (iy)
66     D097 CB23      ctbl3: sla      e
67     D099 3004      jr      nc, ctbl4
68     D09B FDCB00CE      set      1, (iy)
69     D09F CB25      ctbl4: sla      l
70     D0A1 3004      jr      nc, ctbl5
71     D0A3 FDCB00D6      set      2, (iy)
72     D0A7 FD23      ctbl5: inc      iy

```



## リスト2-13 つづき

```

73 D0A9 25          dec    h
74 D0AA 20E3        jr     nz,ctbl2
75 D0AC DD23        inc    ix
76 D0AE DD23        inc    ix
77 D0B0 DD23        inc    ix
78 D0B2 10D0        djnz   ctbl1
79                  ;
80 D0B4 ED738CD1    ld     (save$),sp
81 D0B8 3100CF      ld     sp,stk$top
82                  ;
83 D0BB 2A05D0      ld     hl,(y0)
84 D0BE E5          push   hl
85 D0BF 2A03D0      ld     hl,(x0)
86 D0C2 E5          push   hl
87                  ;
88 D0C3             ; main equ    $
89 D0C3 210000      ld     hl,0
90 D0C6 39          add     hl,$sp      ; hl := sp ( and cy := 0 )
91 D0C7 1100CF      ld     de,$stk$top
92 D0CA ED52        sbc     hl,$de
93 D0CC CAD4D1      jp      z,$endret
94                  ;
95 D0CF C1          pop     bc          ; bc := x
96 D0D0 D1          pop     de          ; de := y
97 D0D1 ED4390D1    ld     (coordx),bc ; save x
98 D0D5 ED5392D1    ld     (coordy),de ; save y
99 D0D9 CDECD1      call    search
100                 ;
101 D0DC CD59D2      call    check
102 D0DF 38E2        jr      c,$main
103                 ;
104 D0E1 CDB4D2      call    hline
105                 ;
106                 ; --- save stack ---
107                 ;
108 D0E4 210000      ld     hl,0
109 D0E7 2284D1      ld     (uf),hl      ; uf,lf off
110 D0EA 2286D1      ld     (cuf),hl
111 D0ED 2A55D2      ld     hl,(x1)
112 D0F0 2253D2      ld     (x),hl      ; x := x1
113 D0F3 3A92D1      ld     a,(coordy)
114 D0F6 B7          or      a          ; if a<>0 then cuf := on
115 D0F7 2808        jr      z,$skip1
116 D0F9 3286D1      ld     (cuf),a
117 D0FC 3D          dec     a
118 D0FD 3288D1      ld     (yminus),a
119 D100 3C          inc     a
120 D101 06C7        skip1: ld     b,199
121 D103 B8          cp      b          ; if a<>199 then clf := on
122 D104 2807        jr      z,$forx
123 D106 3287D1      ld     (clf),a
124 D109 3C          inc     a
125 D10A 328AD1      ld     (yplus),a
126                 ;
127 D10D 3A87D1      forx:  ld     a,(clf)
128 D110 B7          or      a
129 D111 282B        jr      z,$forx2
130 D113 ED5B8AD1    ld     de,(yplus)
131 D117 ED4B53D2    ld     bc,(x)
132 D11B CD1CC5      call    point3
133 D11E DD5600      ld     d,(ix)      ; d := point(x,yplus)
134                 ;
135 D121 3A07D0      ld     a,(bdc)
136 D124 BA          cp      d
137 D125 280A        jr      z,$forx1    ; if d = bdc then "forx1"
138 D127 3A84D1      ld     a,(uf)
139 D12A B7          or      a
140 D12B CC94D1      call    z,$save1      ; if uf = 0 then "save"
141 D12E 3A07D0      ld     a,(bdc)
142 D131 BA          forx1: cp      d
143 D132 200A        jr      nz,$forx2    ; if bdc <> d then "forx2"
144 D134 3A84D1      ld     a,(uf)
145 D137 B7          or      a
146 D138 2804        jr      z,$forx2    ; if uf = 0 then "forx2"
147 D13A AF          xor     a

```

つづく



リスト2-13 つづき

```

148 D13B 3284D1      ld      (uf),a
149                  ;
150 D13E 3A86D1      forx2: ld      a,(cuf)
151 D141 B7          or      a
152 D142 282B        jr      z,forx4
153 D144 ED5B88D1    ld      de,(yminus)
154 D148 ED4B53D2    ld      bc,(x)
155 D14C CD1CC5      call    point3
156 D14F DD5E00      ld      e,(ix)          ; e := point(x,yplus)
157 D152 3A07D0      ld      a,(bdc)
158 D155 BB          cp      e
159 D156 280A        jr      z,forx3          ; if e = bdc then "skip3"
160 D158 3A85D1      ld      a,(lf)
161 D15B B7          or      a
162 D15C CCB1D1      call    z,save2          ; if lf = 0 then "save"
163 D15F 3A07D0      ld      a,(bdc)
164 D162 BB          forx3: cp      e
165 D163 200A        jr      nz,forx4          ; if bdc <> e then "skip4"
166 D165 3A85D1      ld      a,(lf)
167 D168 B7          or      a
168 D169 2804        jr      z,forx4          ; if uf = 0 then "skip2"
169 D16B AF          xor     a
170 D16C 3285D1      ld      (lf),a
171                  ;
172 D16F 2A53D2      forx4: ld      hl,(x)
173 D172 23          inc     hl
174 D173 2253D2      ld      (x),hl
175 D176 ED5B57D2    ld      de,(x2)
176 D17A B7          or      a
177 D17B ED52        sbc     hl,de
178 D17D 388E        jr      c,forx
179 D17F 288C        jr      z,forx
180                  ;
181 D181 C3C3D0      jp      main
182                  ;
183 D184             uf:      defs    1
184 D185             lf:      defs    1
185 D186             cuf:     defs    1
186 D187             clf:     defs    1
187 D188             yminus:  defs    2
188 D18A             yplus:   defs    2
189                  ;
190 D18C             savesp:  defs    2
191 D18E             retadr:  defs    2
192 D190             coordx:  defs    2
193 D192             coordy:  defs    2
194                  ;
195                  ; --- save ---
196                  ;
197 D194 DDE1         save1: pop     ix          ; ix <- return addr.
198 D196 210000      ld      hl,0
199 D199 39          add     hl,sp
200 D19A 0100C8      ld      bc,stkbot
201 D19D B7          or      a
202 D19E ED42        sbc     hl,bc
203 D1A0 382C        jr      c,outstk
204 D1A2 2A8AD1      ld      hl,(yplus)
205 D1A5 E5          push    hl
206 D1A6 2A53D2      ld      hl,(x)
207 D1A9 E5          push    hl
208 D1AA 3EFF        ld      a,0FFH
209 D1AC 3284D1      ld      (uf),a
210 D1AF DDE9        jp      (ix)          ; return
211                  ;
212 D1B1 DDE1         save2: pop     ix          ; ix <- return addr.
213 D1B3 210000      ld      hl,0
214 D1B6 39          add     hl,sp
215 D1B7 0100C8      ld      bc,stkbot
216 D1BA B7          or      a
217 D1BB ED42        sbc     hl,bc
218 D1BD 380F        jr      c,outstk
219 D1BF 2A88D1      ld      hl,(yminus)
220 D1C2 E5          push    hl
221 D1C3 2A53D2      ld      hl,(x)
222 D1C6 E5          push    hl

```

つづく



## リスト2-13 つづき

```

223 D1C7 3EFF          ld      a,0FFH
224 D1C9 3285D1        ld      (lf),a
225 D1CC DDE9          jp      (ix)          ; return
226                    ;
227                    ;
228 D1CE 11DCD1        outstk: ld      de,emes
229 D1D1 CD0B00        call    print
230                    ;
231 D1D4 ED7B8CD1      endret: ld      sp,(savesp)
232 D1D8 2A8ED1        ld      hl,(retadr)
233 D1DB E9            jp      (hl)          ; return
234                    ;
235 D1DC 4F757420      emes:   defm    'Out of Memory'
      D1E0 6F66204D
      D1E4 656D6F72
      D1E8 79
236 D1E9 0D0A00        defb    0DH,0AH,0
237                    ;
238                    ;
239                    ; search color
240                    ;
241                    ;
242                    ;
243 D1EC ED4353D2      search: ld      (x),bc
244 D1F0 2A53D2        left:   ld      hl,(x)
245 D1F3 CB7C          bit      7,h          ; if hl < 0
246 D1F5 201D          jr      nz,left4      ; then x1 := 0
247                    ;
248 D1F7 ED4B53D2      left1:  ld      bc,(x)
249 D1FB ED5B92D1      ld      de,(coordy)
250 D1FF CD1CC5        call    point3
251 D202 3A07D0        ld      a,(bdc)
252 D205 DD4600        ld      b,(ix)          ; b := (pcolor)
253 D208 B8            cp      b          ; if bdc = pcolor
254 D209 2809          jr      z,left4      ; then "left4"
255 D20B 2A53D2        ld      hl,(x)
256 D20E 2B            dec     hl          ; x := x-1
257 D20F 2253D2        ld      (x),hl
258 D212 18DC          jr      left
259                    ;
260 D214 2A53D2      left4:  ld      hl,(x)
261 D217 23            inc     hl          ;
262 D218 2255D2        ld      (x1),hl      ; x1 := x+1
263                    ;
264                    ; right
265                    ;
266 D21B ED4B90D1      right:  ld      bc,(coordx)      ; bc := x
267 D21F ED4353D2      ld      (x),bc
268 D223 2A53D2      right1: ld      hl,(x)
269 D226 117F02        ld      de,639
270 D229 B7            or      a
271 D22A ED52          sbc     hl,de          ; if x > 639
272 D22C 301D          jr      nc,right2      ; then "right2"
273                    ;
274 D22E ED4B53D2      ld      bc,(x)
275 D232 ED5B92D1      ld      de,(coordy)
276 D236 CD1CC5        call    point3
277 D239 3A07D0        ld      a,(bdc)
278 D23C DD4600        ld      b,(ix)          ; if bdc = pcolor
279 D23F B8            cp      b          ; then "right5"
280 D240 2809          jr      z,right2
281 D242 2A53D2        ld      hl,(x)
282 D245 23            inc     hl          ; x := x+1
283 D246 2253D2        ld      (x),hl      ; a := low(x)
284 D249 18D8          jr      right1
285                    ;
286 D24B 2A53D2      right2: ld      hl,(x)
287 D24E 2B            dec     hl
288 D24F 2257D2        ld      (x2),hl
289                    ;
290 D252 C9            ret
291                    ;
292 D253              x:      defs    2
293 D255              x1:     defs    2

```

つづく



リスト2-13 つづき

```

294 D257      x2:      defs      2
295          :
296          :      check already painted
297          :
298          :      entry      x1
299          :      x2
300          :      return      cy flag      (if on then painted else no)
301          :      tileno
302 D259 2A55D2  check: ld      hl,(x1)
303 D25C 2253D2      ld      (x),hl      ; x := x1
304 D25F 3A08D0      ld      a,(no)
305 D262 57          ld      d,a
306 D263 3A92D1      ld      a,(coordy)      ; a := low(coordy)
307 D266 92          check1: sub      d
308 D267 30FD      jr      nc,check1
309 D269 82          add      a,d      ; a := y mod no
310 D26A 2112D0      ld      hl,tilec
311 D26D 010800      ld      bc,8
312 D270 3C          inc      a
313 D271 3D          check2: dec      a
314 D272 2803      jr      z,check3
315 D274 09          add      hl,bc
316 D275 18FA      jr      check2
317 D277 22B2D2  check3: ld      (tileno),hl
318          ;
319 D27A FD2AB2D2  check4: ld      iy,(tileno)
320 D27E ED4B53D2      ld      bc,(x)
321 D282 79          ld      a,c
322 D283 E607      and      7      ; a := x mod 8
323 D285 08          ex      af,af'      ; save a
324 D286 ED5B92D1      ld      de,(coordy)
325 D28A CD1CC5      call     point3
326 D28D DD4600      ld      b,(ix)      ; b := point(x,y)
327 D290 08          ex      af,af'      ; load a
328 D291 5F          ld      e,a
329 D292 1600      ld      d,0
330 D294 FD19      add      iy,de      ; get tile color code
331 D296 FD7E00      ld      a,(iy)
332 D299 B8          cp      b      ; if point(x,y)<>tilecolor
333 D29A 2014      jr      nz,check5
334 D29C 2A53D2      ld      hl,(x)
335 D29F 23          inc      hl
336 D2A0 2253D2      ld      (x),hl      ; x := x+1
337 D2A3 ED5B57D2      ld      de,(x2)
338 D2A7 B7          or      a
339 D2A8 ED52      sbc      hl,de      ; if x <= x2
340 D2AA 38CE      jr      c,check4      ; then "check4"
341 D2AC 28CC      jr      z,check4
342          ;
343 D2AE 37          scf      ; cy := 1
344 D2AF C9          ret
345          ;
346 D2B0 B7          check5: or      a      ; cy := 0
347 D2B1 C9          ret
348          ;
349 D2B2      tileno: defs      2
350          :
351          :
352          :      --- hline ---
353          :      entry      x1,x2,coordy
354          :      tileno
355          :      return      none
356          :
357 D2B4 2A55D2  hline: ld      hl,(x1)
358 D2B7 2253D2      ld      (x),hl      ; x := x1
359 D2BA DD2AB2D2      ld      ix,(tileno)      ; ix <- tile color
360          ;
361 D2BE DDE5      hline1: push     ix
362 D2C0 3A53D2      ld      a,(x)      ; a := low(x)
363 D2C3 E607      and      7
364 D2C5 4F          ld      c,a
365 D2C6 0600      ld      b,0
366 D2C8 DD09      add      ix,bc
367 D2CA DD7E00      ld      a,(ix)      ; a <- color code

```

つづく



## リスト2-13 つづき

368	D2CD	3207C1	ld	(color),a	
369	D2D0	ED4B53D2	ld	bc,(x)	
370	D2D4	ED5B92D1	ld	de,(coordy)	
371	D2D8	CD1DC1	call	pset3	
372	D2DB	DDE1	pop	ix	
373	D2DD	2A53D2	ld	hl,(x)	; x := x+1
374	D2E0	23	inc	hl	
375	D2E1	2253D2	ld	(x),hl	
376	D2E4	ED5B57D2	ld	de,(x2)	; if x <= x2
377	D2E8	B7	or	a	; then "hline1"
378	D2E9	ED52	sbc	hl,de	
379	D2EB	38D1	jr	c,hline1	
380	D2ED	28CF	jr	z,hline1	
381	D2EF	C9	ret		







---

## 3-1 効果音

---

## 3-2 音声合成

---

PSG(AY-3-8910)は、1チップのシンセサイザと呼ばれていますが、機能的にはシンセサイザと呼べるほどのものではありません。しかし、ソフトウェア次第でなかなか面白いことができます。本章ではそれらの1例を紹介します。



3-1

効果音

本章では、PSG の機能、レジスタなどについてはいっさい解説しません。これに関しては、既刊の『X 1 リファレンスノウト』などを参照してください。

3-1-1

基本サブルーチン

PSG にデータを出力する最も基本となるサブルーチンをリスト 3-1 に示します。これは既に何度も見てきたと思うので、特に解説はしません。

PSG で音楽を演奏する場合、トーン・ジェネレータに音階の周波数をセットし、音の長さはソフトウェアでその分カウントしてやります。リスト 3-2 は音楽演奏用のサブルーチンで、音の長さには表 3-1 のようなデータを入れます。音の高さのデータは、音階の周波数から求めたもので、これについては『X 1 リファレンスノウト』を参照してください。

リスト3-1

1	:		
2	:	---	subroutine 'sound' ---
3	:	entry	a := psg reg. number
4	:		d := data
5	:	modifies	none
6	:		
7	1C00	psgcom equ	1C00H
8		:	
9	C000	org	0C000H
10		:	
11	C000 C5	sound: push	bc
12	C001 01001C	ld	bc,psgcom
13	C004 ED79	out	(c),a
14	C006 05	dec	b
15	C007 ED51	out	(c),d
16	C009 C1	pop	bc
17	C00A C9	ret	



## リスト3-2

```

1      ;
2      ; ---      music      ---
3      ;          entry      none
4      ;          modifies    af, bc, de, hl
5      ;
6 C000  sound      equ      0C000H
7      ;
8 C100  ;          org      0C100H
9      ;
10 C100 2107C1  music: ld      hl, mdata
11 C103 CD29C1      call     music1
12 C106 C9          ret
13      ;
14 C107 EF00  mdata: defw     0EFH          ; freq.
15 C109 8000      defw     0080H         ; length
16 C10B D500      defw     0D5H
17 C10D 8000      defw     0080H
18 C10F BE00      defw     0BEH
19 C111 8000      defw     0080H
20 C113 B300      defw     0B3H
21 C115 8000      defw     0080H
22 C117 9F00      defw     9FH
23 C119 8000      defw     0080H
24 C11B 8E00      defw     8EH
25 C11D 8000      defw     0080H
26 C11F 7F00      defw     7FH
27 C121 8000      defw     0080H
28 C123 7700      defw     77H
29 C125 8000      defw     0080H
30      ;
31      ;
32      ;
33      ;
34 C127 FFFF      defw     0FFFFH          ; end mark!
35      ;
36 C129 3E07  music1: ld      a, 7          ; turn switch
37 C12B 163E      ld      d, 3EH          ; tone-a on
38 C12D CD00C0      call     sound
39 C130 3E08  music2: ld      a, 8          ; set volume
40 C132 1600      ld      d, 0
41 C134 CD00C0      call     sound
42 C137 4E      ld      c, (hl)          ; get freq. data
43 C138 23      inc     hl
44 C139 46      ld      b, (hl)
45 C13A 23      inc     hl
46 C13B 78      ld      a, b
47 C13C A7      and     a          ; freq. < 0 ?
48 C13D F8      ret     m          ; yes . return
49 C13E B1      or      c          ; freq.=0 ?
50 C13F 280A      jr      z, music3      ; yes . volume := 0
51 C141 3E08      ld      a, 8          ; volume := 8
52 C143 160F      ld      d, 15
53 C145 CD00C0      call     sound
54 C148 CD54C1      call     tone
55 C14B 4E  music3: ld      c, (hl)          ; get length data
56 C14C 23      inc     hl
57 C14D 46      ld      b, (hl)
58 C14E 23      inc     hl
59 C14F CD5FC1      call     delay
60 C152 18DC      jr      music2
61      ;
62      ; set freq.
63      ;
64 C154 AF  tone:  xor     a          ; set freq.
65 C155 51      ld      d, c

```

つづく



リスト3-2 つづき

66	C156	CD00C0	call	sound
67	C159	3C	inc	a
68	C15A	50	ld	d,b
69	C15B	CD00C0	call	sound
70	C15E	C9	ret	
71			:	
72			:	count down
73			:	
74	C15F	112601	delay:	ld de,0126H
75	C162	1B	delay2:	dec de
76	C163	7B	ld	a,e
77	C164	B2	or	d
78	C165	20FB	jr	nz,delay2
79	C167	0B	dec	bc
80	C168	79	ld	a,c
81	C169	B0	or	b
82	C16A	20F3	jr	nz,delay
83	C16C	C9	ret	

表3-1 音の長さのデータ

音 符											
デ ー タ	400H	300H	200H	180H	100H	C0H	80H	60H	40H	30H	20H

♪ = 120の場合、♪ = nの場合はこのデータを $\frac{120}{n}$ 倍する

## 3-1-2 ノイズ・ジェネレータの活用

効果音をうまく作るコツはノイズ・ジェネレータを活用することです。簡単な例として、波の音を作ってみました（リスト3-3）。「sounds」というラベル名で始まるサブルーチンには、インライン・パラメータでデータを送ります。データの形式は、PSGレジスタ番号、データ……の順です。

リスト3-4にいくつかのサンプルのデータを示しておきます。最初の踏切の音は、3つの周波数の違うトーンを出力し、そのうち2つにエンベロープをかけて作りました。次はピストルの発射音ですが、エンベロープ・タイムをもっと短くして、エンベロープ波形を連続したものに取り替えるとマシンガンの音になります。3番目のデータは爆発音ですが、これは普通のシンセサイザでは、ピンク・ノイズを使って合



成します。ピンク・ノイズは高周波成分の少ないノイズですが、X 1ではPSGへの基本周波数が高く、ノイズの平均周波数は最低でも4 KHzと高くなっています。そのためここではトーンに最も低い周波数をセットして、同じチャンネルからミックスして出力しました。

リスト 3-5 はサイレンの音をです。これは2つの周波数のトーンを交互に出力することで合成できます。

リスト3-3

```

1      ;
2      ; --- effects ---
3      ;      entry    none
4      ;      modifies af.de.hl
5      ;
6 C000      sound    equ    0C000H
7      ;
8 C200      ;      org    0C200H
9      ;
10 C200 CD11C2  effect: call    sounds
11      ; --- wave ---
12 C203 0737      defb    7,37H      ; noise-a on
13 C205 0608      defb    6,08H      ; noise freq. := 15.6KHz
14 C207 0810      defb    8,10H      ; env.-a on
15 C209 0B7A      defb    11,7AH     ; env.time := 8s
16 C20B 0CDA      defb    12,0DAH
17 C20D 0D0E      defb    13,0EH     ; env.shape := 14
18      ;
19      ;
20      ;
21      ;
22 C20F FF      defb    0FFH      ; end mark!
23 C210 C9      ret
24      ;
25 C211 E1      sounds: pop    hl      ; hl := addr. of data
26 C212 7E      sloop: ld      a,(hl)  ; a := reg. no.
27 C213 23      inc    hl
28 C214 A7      and     a
29 C215 FA1FC2  jp      m,exit      ; if reg.no < 0 then return
30 C218 56      ld      d,(hl)      ; d := data
31 C219 23      inc    hl
32 C21A CD00C0  call    sound
33 C21D 18F3      jr      sloop
34 C21F E9      exit:  jp      (hl)      ; return

```



リスト3-4

```

1 ; --- fumikiri ---
2     defb 7,38H           ; tone_a,b,c on
3     defb 8,10H           ; envelope_a on
4     defb 9,10H           ; envelope_b on
5     defb 10,0CH          ; volume_c := 12
6     defb 0,77H           ; tone_a freq. := 1047Hz
7     defb 1,0             ;
8     defb 2,0EEH          ; tone_b freq. := 523Hz
9     defb 3,0             ;
10    defb 4,0DDH          ; tone_c freq. := 262Hz
11    defb 5,01H           ;
12    defb 11,42H          ; envelope time := 0.5s
13    defb 12,0FH          ;
14    defb 13,08H          ; envelope shape := 8
15    defb 0FFH            ; end mark!
16 ;
17 ; --- pistol ---
18    defb 7,37H           ; noise_a on
19    defb 6,8              ; noise freq. := 15KHz
20    defb 8,10H           ; envelope_a on
21    defb 11,42H          ; envelope time := 0.5s
22    defb 12,02H          ;
23    defb 13,0            ; envelope shape := 0
24    defb 0FFH            ; end mark !
25 ;
26 ; --- explosion ---
27    defb 7,36H           ; tone_a on , noise_a on
28    defb 8,10H           ; envelope_a on
29    defb 0,0FFH          ; tone_a freq. := 30.5Hz
30    defb 1,0FH           ;
31    defb 6,1FH           ; noise freq. := 4KHz
32    defb 11,8EH          ; envelope time := 3s
33    defb 12,5BH          ;
34    defb 13,0            ; envelope shape := 0
35    defb 0FFH            ; end mark!

```

リスト3-5

```

1      ;
2      ; --- siren ---
3      ;
4 C100      sounds equ 0C100H
5      ;
6 C300      org 0C300H
7      ;
8 C300 CD00C1 siren: call sounds
9 C303 073C      defb 7,3CH           ; tone_a on , tone_b on
10 C305 0810      defb 8,10H          ; envelope_a on
11 C307 0910      defb 9,10H          ; envelope_b on
12 C309 0BFF      defb 11,0FFH        ; envelope time := 8s
13 C30B 0CF4      defb 12,0F4H        ;
14 C30D 0D0E      defb 13,0EH          ; envelope shape := 14
15 C30F FF        defb 0FFH           ; end mark!
16      ;
17 C310 1E32      ld e,50              ; loop counter
18 C312 CD00C1 siren1: call sounds
19 C315 00EF      defb 0,0EFH          ; tone_a freq. := 523Hz
20 C317 0100      defb 1,0            ;
21 C319 028E      defb 2,8EH          ; tone_b freq. := 880Hz
22 C31B 0300      defb 3,0            ;
23 C31D FF        defb 0FFH           ; end mark!
24      ;
25 C31E CD34C3      call delay

```



リスト3-5 つづき

```
26 C321 CD00C1      call    sounds
27 C324 00DD          defb    0,0DDH      ; tone_a freq. := 262Hz
28 C326 0101          defb    1,01H
29 C328 021C          defb    2,1CH      ; tone_b freq. := 440Hz
30 C32A 0301          defb    3,1
31 C32C FF            defb    0FFH      ; end mark!
32
33 C32D CD34C3      call    delay
34 C330 1D            dec     e
35 C331 20DF          jr      nz,siren1
36 C333 C9            ret
37
38 C334 01FFFF      : delay: ld      bc,0FFFFH
39 C337 0B            delay1: dec    bc
40 C338 79            ld      a,c
41 C339 B0            or      b
42 C33A 20FB          jr      nz,delay1
43 C33C C9            ret
```

### 3-1-3 特殊効果音

PSG はシンセサイザとして見れば非常にシンプルなもの  
で、音はどうしても単調になりがちです。しかし、そういっ  
てあきらめるのも早計で、プログラムの工夫次第で結構おも  
しろい効果が得られます。ここでは一般的な効果音としてポ  
ルタメント、エコー、ビブラートを紹介します。

●急降下爆撃

トーン・ジェネレータの周波数を高い方から低い方へ変化  
させポルタメントの効果を得ています（リスト 3-6）。3 つ  
のトーンを合成すればよりスピーシーで広がりのある音が得  
られます。

●エコー

音が物体に反射して帰ってくると、そこに時間の遅れを生  
じエコー効果が得られます。この遅れをソフトウェアで作っ  
てやれば、同様の効果が得られるはずです。

ここでは PSG の 3 つのトーン・ジェネレータを利用して  
同じ音を少しずつ遅らせて演奏しています。もちろん、元の  
音に比べて反射して来た音は小さいはずなので、音量もだん  
だん小さくしなければなりません。リスト 3-7 の‘echo 1’で  
始まるサブルーチンでは、この遅れ時間と音量を自由に設定  
できるようにしています。実際、お風呂で歌うとうまく聞こ



えるように、このプログラムを使ったときの効果は大きく、単調な PSG の音がかなり艶やかに聞こえます。

### ●ビブラート

エレクトーンでおなじみのビブラートの効果は、音に低い周波数で FM 変調をかけると得られます。ビブラートをかけるときにはこの低い周波数を発生する LFO (Low Frequency Oscillator) を使うのですが、PSG にはこの LFO に相当する機能がないので、これをソフトウェアの方で作ります。

FM 変調というちょっと面倒くさそうですが、実際にはトーン・ジェネレータのデータを一定の周期で増減させるだけで基本的に難しいことはありません。ここではビブラートの深さと周波数を調節できるようにしたため、一定の周波数を作り出す部分（本来は LFO が担当する）の記述が若干長くなりました（リスト 3-8）。

リスト 3-6

```
1      ;
2      ; --- bomb ---
3      ;
4 C000  sound equ 0C000H
5 C100  sounds equ 0C100H
6      ;
7 C400  org 0C400H
8      ;
9 C400 CD00C1 bomb: call sounds
10 C403 073E      defb 7,3EH      ; tone_a on
11 C405 080F      defb 8,0FH      ; volume_a := 15
12 C407 FF        defb 0FFH      ; end mark!
13 C408 210600    ld hl,6        ; tone_a freq. := 20.8KHz
14      ;
15      ; 20.8KHz -> 244.6Hz
16      ;
17 C40B AF bomb1: xor a
18 C40C 55        ld d,l        ; hl := data
19 C40D CD00C0    call sound
20 C410 3C        inc a
21 C411 54        ld d,h
22 C412 CD00C0    call sound
23 C415 CD33C4    call delay
24 C418 23        inc hl
25 C419 7C        ld a,h
26 C41A E6FE      and 0FEH      ; freq. := 244.6Hz ?
27 C41C 28ED      jr z,bomb1    ; no , continue
28 C41E CD00C1    call sounds
29 C421 0736      defb 7,36H    ; tone_a on , noise_a on
30 C423 00FF      defb 0,0FFH  ; tone_a freq. := 30.5Hz
31 C425 010F      defb 1,0FH
32 C427 061F      defb 6,1FH   ; noise freq. := 4KHz
33 C429 0810      defb 8,10H   ; envelope_a on
34 C42B 0B24      defb 11,24H  ; envelope time := 8s
35 C42D 0CF4      defb 12,0F4H
```

つづく



## リスト3-6 つづき

```

36 C42F 0D00      defb      13,0          ; envelope shape := 0
37 C431 FF        defb      0FFH          ; end mark!
38 C432 C9        ret
39                ;
40 C433 010004    delay: ld      bc,0400H
41 C436 0B        delay1: dec     bc
42 C437 79        ld      a,c
43 C438 B0        or      b
44 C439 20FB      jr      nz,delay1
45 C43B C9        ret

```

## リスト3-7

```

1      :
2      : ---      echo      ---
3      :          entry      none
4      :          modifies   af, bc, de, hl, ix, iy
5      :
6 C000  sound      equ      0C000H
7      :
8 C500  org      0C500H
9      :
10 C500 2107C5    echo:  ld      hl, mdata
11 C503 CD29C5    call    echol
12 C506 C9        ret
13      :
14 C507 EF00      mdata: defw     0EFH          : freq.
15 C509 0001      defw     0100H          : length
16 C50B D500      defw     0D5H
17 C50D 0001      defw     0100H
18 C50F BE00      defw     0BEH
19 C511 0001      defw     0100H
20 C513 B300      defw     0B3H
21 C515 0001      defw     0100H
22 C517 9F00      defw     9FH
23 C519 0001      defw     0100H
24 C51B 8E00      defw     8EH
25 C51D 0001      defw     0100H
26 C51F 7F00      defw     7FH
27 C521 0001      defw     0100H
28 C523 7700      defw     77H
29 C525 0001      defw     0100H
30      :
31      :
32      :
33      :
34 C527 FFFF      defw     0FFFFH          ; end mark!
35      :
36 C529          echol   equ      $
37 C529 CD6EC5    call    iniech
38 C52C 112601    echo2: ld      de, 0126H          : delay
39 C52F 1B        echo3: dec     de
40 C530 7B        ld      a, e
41 C531 B2        or      d
42 C532 20FB      jr      nz, echo3
43      :
44 C534 DD21FFC5  ld      ix, voctbl          : ix := voice table
45 C538 0603      ld      b, 3
46 C53A 3E03      echo4: ld      a, 3
47 C53C 90        sub     b
48 C53D 320BC6    ld      (voice), a
49 C540 DD5E00      ld      e, (ix+count)          : de := counter
50 C543 DD5601      ld      d, (ix+count+1)
51 C546 DD7E01      ld      a, (ix+count+1)
52 C549 A7        and     a          : finish ?
53 C54A F252C5    jp      p, echo5          : no, continue
54 C54D 78        ld      a, b

```

つづく



リスト3-7 つづき

```

55 C54E 3D          dec    a          ; finish all ?
56 C54F C8          ret     z          ; yes , return
57 C550 1813        jr      echo6      ; next voice
58                  ;
59 C552 DDB600      echo5: or      (ix+count) ; count down complete ?
60 C555 CCA5C5      call    z,rdnext    ; read next data
61 C558 DD5E00      ld      e,(ix+count) ; count down
62 C55B DD5601      ld      d,(ix+count+1)
63 C55E 1B          dec     de
64 C55F DD7300      ld      (ix+count),e
65 C562 DD7201      ld      (ix+count+1),d
66                  ;
67 C565 110400      echo6: ld      de,4          ; count down all voice ?
68 C568 DD19        add     ix,de
69 C56A 10CE        djnz    echo4      ; no , continue
70 C56C 18BE        jr      echo2
71                  ;
72                  ; initialize echo
73                  ;
74 C56E 3E07        iniech: ld      a,7          ; turn switch
75 C570 1638        ld      d,38H        ; tone_a,b,c on
76 C572 CD00C0      call    sound
77 C575 0603        ld      b,3          ; volume_a,b,c := 0
78 C577 3E08        ld      a,8
79 C579 1600        ld      d,0
80 C57B CD00C0      ini1:  call    sound
81 C57E 3C          inc     a
82 C57F 10FA        djnz    ini1
83 C581 3E03        ld      a,3          ; set voice counter
84 C583 DD21FFC5    ld      ix,voctbl    ; ix := voice table
85 C587 01F6C5      ld      bc,dlytim    ; bc := delay time table
86 C58A 110400      ld      de,4
87 C58D F5          ini2:  push    af
88 C58E 0A          ld      a,(bc)        ; set delay time
89 C58F 03          inc     bc
90 C590 DD7700      ld      (ix+count),a
91 C593 0A          ld      a,(bc)
92 C594 03          inc     bc
93 C595 DD7701      ld      (ix+count+1),a
94 C598 DD7502      ld      (ix+point),l        ; set counter
95 C59B DD7403      ld      (ix+point+1),h
96 C59E DD19        add     ix,de        ; initialize all voice ?
97 C5A0 F1          pop     af
98 C5A1 3D          dec     a
99 C5A2 20E9        jr      nz,ini2      ; no , continue
100 C5A4 C9         ret
101                  ;
102                  ; read next data
103                  ;
104 C5A5 3A0BC6      rdnext: ld      a,(voice)
105 C5A8 C608        add     a,8
106 C5AA 1600        ld      d,0          ; volume := 0
107 C5AC CD00C0      call    sound
108 C5AF DD6E02      ld      l,(ix+point)    ; hl := data pointer
109 C5B2 DD6603      ld      h,(ix+point+1)
110 C5B5 56          ld      d,(hl)        ; get freq. data
111 C5B6 23          inc     hl
112 C5B7 5E          ld      e,(hl)
113 C5B8 23          inc     hl
114 C5B9 7B          ld      a,e
115 C5BA A7          and     a          ; freq. < 0 ?
116 C5BB FAE9C5      jp      m,rd2          ; yes , do nothing
117 C5BE B2          or      d          ; freq. = 0 ?
118 C5BF 2824        jr      z,rd1          ; yes , volume := 0
119 C5C1 3A0BC6      ld      a,(voice)    ; set freq.
120 C5C4 CB27        sla     a
121 C5C6 CD00C0      call    sound
122 C5C9 3C          inc     a
123 C5CA 53          ld      d,e
124 C5CB CD00C0      call    sound
125                  ;
126 C5CE 3A0BC6      ld      a,(voice)    ; calc. adr. of volume data
127 C5D1 5F          ld      e,a
128 C5D2 1600        ld      d,0
129 C5D4 FD21FCC5    ld      iy,volume

```

つづく



## リスト3-7 つづき

```

130 C5D8 FD19          add    iy,de
131 C5DA FD5600        ld      d,(iy)          ; get volume data
132 C5DD 3A0BC6        ld      a,(voice)       ; set volume
133 C5E0 C608          add     a,8
134 C5E2 CD00C0        call    sound
135 C5E5 5E            rd1:   ld      e,(hl)          ; get length data
136 C5E6 23            inc     hl
137 C5E7 56            ld      d,(hl)
138 C5E8 23            inc     hl
139 C5E9 DD7300        rd2:   ld      (ix+count),e    ; set counter
140 C5EC DD7201        ld      (ix+count+1),d
141 C5EF DD7502        ld      (ix+point),l    ; save data pointer
142 C5F2 DD7403        ld      (ix+point+1),h
143 C5F5 C9            ret
144                    ;
145                    ; data for echo ( delay time & volume )
146                    ;
147 C5F6 0000        dlytim: defw    0000H          ; voice_a delay time
148 C5F8 5000        defw    0050H          ; b
149 C5FA A000        defw    00A0H          ; c
150 C5FC 0F        volume: defb    0FH          ; voice_a volume
151 C5FD 0B        defb    0BH          ; b
152 C5FE 07        defb    07H          ; c
153                    ;
154 0000        count    equ    0
155 0002        point    equ    2
156                    ;
157 C5FF        voctbl:  defs    12
158 C60B        voice:   defs    1
159                    ;

```

## リスト3-8

```

1                    ;
2                    ; --- vibrato ---
3                    ;
4                    ; entry      none
5                    ; modifies   af, bc, de, hl
6 C000        sound    equ    0C000H
7                    ;
8 C600        org      0C600H
9                    ;
10 C600 2107C6    vibrat: ld      hl, mdata
11 C603 CD29C6    call    vib1
12 C606 C9        ret
13                    ;
14 C607 EF00        mdata:  defw    0EFH          ; freq.
15 C609 0001        defw    0100H          ; length
16 C60B D500        defw    0D5H
17 C60D 0001        defw    0100H
18 C60F BE00        defw    0BEH
19 C611 0001        defw    0100H
20 C613 B300        defw    0B3H
21 C615 0001        defw    0100H
22 C617 9F00        defw    9FH
23 C619 0001        defw    0100H
24 C61B 8E00        defw    8EH
25 C61D 0001        defw    0100H
26 C61F 7F00        defw    7FH
27 C621 0001        defw    0100H
28 C623 7700        defw    77H
29 C625 0001        defw    0100H
30                    ;
31                    ;
32                    ;
33                    ;
34 C627 FFFF        defw    0FFFFH          ; end mark!
35                    ;
36 C629          vib1    equ    $
37 C629 3E07        ld      a,7          ; turn switch

```

つづく



リスト3-8 つづき

```

38 C62B 163E          ld      d,3EH          ; tone_a on
39 C62D CD00C0        call     sound
40 C630 3E08          vib2:  ld      a,8          ; set volume
41 C632 1600          ld      d,0          ; volume := 0
42 C634 CD00C0        call     sound
43 C637 4E            ld      c,(hl)        ; get freq. data
44 C638 23            inc      hl
45 C639 46            ld      b,(hl)
46 C63A 23            inc      hl
47 C63B ED43CBC6      ld      (tnfreq),bc    ; save freq. data
48 C63F 78            ld      a,b
49 C640 A7            and      a          ; freq. < 0 ?
50 C641 F8            ret      m          ; yes , return
51 C642 B1            or      c          ; freq. = 0 ?
52 C643 2811          jr      z,vib3        ; yes , volume := 0
53 C645 3E08          ld      a,8          ; set volume
54 C647 160F          ld      d,15         ; volume := 8
55 C649 CD00C0        call     sound
56                   ;
57 C64C AF            xor      a          ; set freq.
58 C64D 51            ld      d,c
59 C64E CD00C0        call     sound
60 C651 3C            inc      a
61 C652 50            ld      d,b
62 C653 CD00C0        call     sound
63 C656 4E            vib3:  ld      c,(hl)        ; get length data
64 C657 23            inc      hl
65 C658 46            ld      b,(hl)
66 C659 23            inc      hl
67 C65A E5            push     hl
68 C65B CD61C6        call     cntdwn        ; count down
69 C65E E1            pop      hl
70 C65F 18CF          jr      vib2
71                   ;
72                   ; count down
73                   ;
74 C661 CD9DC6        cntdwn: call     inilfo        ; initialize LFO
75 C664 112601        cnt1:  ld      de,0126H      ; delay
76 C667 1B            cnt2:  dec      de
77 C668 7B            ld      a,e
78 C669 B2            or      d
79 C66A 20FB          jr      nz,cnt2
80                   ;
81 C66C 0B            dec      bc          ; count down
82 C66D 3ACFC6        ld      a,(frqlfo)
83 C670 A1            and      c          ; oscillate low freq. ?
84 C671 20F1          jr      nz,cnt1        ; no , continue
85 C673 2ACBC6        ld      hl,(tnfreq)      ; hl := current freq.
86 C676 ED5BCDC6      ld      de,(frqmod)      ; de := freq. modulation
87 C67A 3AD0C6        ld      a,(udflg)      ; check freq. up/down
88 C67D 3C            inc      a
89 C67E 32D0C6        ld      (udflg),a
90 C681 E603          and      03H
91 C683 EA8AC6        jp      pe,cnt3        ; up , then tnfreq+frqmod
92 C686 ED52          sbc      hl,de        ; down , then tnfreq-frqmod
93 C688 1801          jr      cnt4
94 C68A 19            cnt3:  add      hl,de
95 C68B 22CBC6        cnt4:  ld      (tnfreq),hl    ; save freq.
96                   ;
97 C68E AF            xor      a          ; set freq.
98 C68F 55            ld      d,1
99 C690 CD00C0        call     sound
100 C693 3C           inc      a
101 C694 54           ld      d,h
102 C695 CD00C0        call     sound
103                   ;
104 C698 79           ld      a,c          ; complete
105 C699 B0           or      b
106 C69A 20C8          jr      nz,cnt1
107 C69C C9           ret
108                   ;
109                   ; initialize LFO
110                   ;
111 C69D AF           inilfo: xor      a

```

つづき



## リスト3-8 つづき

```

112 C69E 32D0C6      ld      (udflg),a      ; clear udflg
113 C6A1 ED5BCBC6    ld      de,(tnfreq)    ; calc. modulation data
114 C6A5 3ACAC6      ld      a,(vibdpt)
115 C6A8 D60C        sub      12
116 C6AA 2807        inilf1: jr      z,inilf2
117 C6AC CB3A        srl      d
118 C6AE CB1B        rr      e
119 C6B0 3C          inc      a
120 C6B1 18F7        jr      inilf1
121 C6B3 ED53CDC6    inilf2: ld      (frqmod),de    ; set modulation data
122 C6B7 1EFF        ld      e,0FFH      ; calc. LFO freq.
123 C6B9 3AC9C6      ld      a,(frqrat)
124 C6BC A7          and      a
125 C6BD 2805        inilf3: jr      z,inilf4
126 C6BF CB3B        srl      e
127 C6C1 3D          dec      a
128 C6C2 18F9        jr      inilf3
129 C6C4 7B          inilf4: ld      a,e
130 C6C5 32CFC6      ld      (frqlfo),a    ; set freq. data
131 C6C8 C9          ret
132                  ;
133                  ;      data for LFO & work area
134                  ;
135 C6C9 04          frqrat: defb      4
136 C6CA 06          vibdpt: defb      6
137                  ;
138 C6CB            tnfreq: defs      2
139 C6CD            frqmod: defs      2
140 C6CF            frqlfo: defs      1
141 C6D0            udflg:  defs      1
142                  ;

```



3-2

音声合成

入力されてきた音声データにしたがって PSG の出力を ON / OFF すると音声出力ができます。実際に音声合成の原理を説明するとかなりの量になるのでここでは省略しますが、プログラムの方は実に簡単です（リスト 3-9）。

リスト3-9

```

1      ;
2      ; --- voice ---
3      ;
4      ; entry      none
5      ; modifies   af,d
6      ; -- IOCS --
7 0DEC  cmtcom equ 0DECH      ; cassette control
8 004A  brkchk equ 004AH     ; sense shift+break
9      ;
10 0002  cmtply equ 02H      ; cmt play
11 0001  cmtstp equ 01H      ; cmt stop
12 1A01  ppi equ 1A01H      ; 8255_2 port_b
13      ;
14 C000  sound equ 0C000H
15      ;
16 C800      org 0C800H
17 C800 AF   xor a           ; tone_a freq. := 0
18 C801 1600 ld d,0
19 C803 CD00C0 call sound
20 C806 3C   inc a
21 C807 CD00C0 call sound
22 C80A 3E07 ld a,7
23 C80C 163E ld d,3EH      ; tone_a on
24 C80E CD00C0 call sound
25      ;
26 C811 01011A ld bc,ppi
27 C814 3E02 ld a,cmtply    ; play
28 C816 CDEC0D call cmtcom
29      ;
30 C819 CD4A00 loop: call brkchk
31 C81C 2811 jr z,break
32 C81E ED78 in a,(c)      ; read data
33 C820 E602 and 2
34 C822 1608 ld d,8        ; volume := 8
35 C824 2002 jr nz,skip    ; if data = 0 then
36 C826 1600 ld d,0        ; volume := 0
37 C828 3E08 skip: ld a,8   ; set volume
38 C82A CD00C0 call sound
39 C82D 18EA jr loop
40      ;
41 C82F 3E01 break: ld a,cmtstp ; stop
42 C831 CDEC0D call cmtcom
43 C834 C9   ret
44      ;

```

これは 8255 ②のポート B のビット 2 からカセットのデータを読み込み、このデータにしたがって単純に PSG を ON



／OFF しています。しかし、カセット内臓タイプのX 1ではデータの信頼性のためかなり波形を整形してくるので、あまりいい音が出ません。同様のプログラムをMSXやFM 7などで実行するとかなり明瞭に聞こえるのですが…。また、オーケストラのように周波数帯域の広いものには向いておらず、せいぜい話し声程度しか再生できません。

リスト 3-10 は、音のデータをメモリに記憶、再生するプログラムです。記憶するときは1ビットのデータを1バイトになるまで繰り返し、それをメモリに入れています。逆に再生するときは1バイトのデータから1ビットずつ取り出しPSGをON／OFFしています。83行目のaレジスタに入れるディレイ・カウンタの値を大きくすると記憶できる音の長さが増しますが、逆に音質は低下します。このプログラムで声のデータを収集し、X 1にしゃべらせたりすると、結構受ける(?) かもしれません。

リスト3-10

```
1      ;
2      ; --- record & play ---
3      ;                               entry      none
4      ;                               modifies   af.b.de.hl.ix
5      ;
6      ; -- IOCS --
7 0DEC  cmtcom equ 0DECH           ; cassette control
8 004A  brkchk equ 004AH          ; sense shift+break
9      ;
10 0002  cmtply equ 02H            ; cmt play
11 0001  cmtstp equ 01H           ; cmt stop
12 1A01  ppi equ 1A01H            ; 8255_2 port_b
13      ;
14 CA00  buffer equ 0CA00H
15 3501  bfsz equ 0FF00H-buffer+1
16      ;
17 C000  sound equ 0C000H
18      ;
19 C900      org 0C900H
20      ;
21      ; record
22      ;
23 C900 3E02      ld a,cmtply      ; cmt play
24 C902 CDEC0D    call cmtcom
25      ;         call loop
26 C905 DD2100CA  ld ix,buffer
27 C909 210135    ld hl,bfsz
28 C90C 01011A    ld bc,ppi
29 C90F 1608      rec1: ld d,8      ; loop counter
30 C911 CD6CC9    rec2: call delay
```

つづく



リスト3-10 つづき

```

31 C914 ED78      in      a,(c)      ; read data
32 C916 E602      and      2
33 C918 1F        rra
34 C919 1F        rra
35 C91A CB1B      rr      e      ; e := data
36 C91C 15        dec      d
37 C91D 20F2      jr      nz,rec2
38 C91F DD7300    ld      (ix),e
39 C922 DD23      inc      ix
40 C924 2B        dec      hl
41 C925 7C        ld      a,h
42 C926 B5        or      l      ; buffer full ?
43 C927 20E6      jr      nz,rec1  ; no , continue
44                ;
45 C929 3E01      ld      a,cmtstp  ; cmt stop
46 C92B CDEC0D    call    cmtcom
47 C92E C331C9    jp      play
48                ;
49                ; play
50                ;
51 C931 AF        play:  xor      a
52 C932 1600      ld      d,0      ; tone_a freq. := 0
53 C934 CD00C0    call    sound
54 C937 3C        inc      a
55 C938 CD00C0    call    sound
56 C93B 3E07      ld      a,7
57 C93D 163E      ld      d,3EH   ; tone_a on
58 C93F CD00C0    call    sound
59                ;
60 C942 DD2100CA  ld      ix,buffer
61 C946 210135    ld      hl,bfsize
62 C949 CD4A00    play1: call    brkchk  ; break ?
63 C94C C8        ret      z      ; yes , return
64                ;
65 C94D DD5E00    ld      e,(ix)
66 C950 0608      ld      b,8      ; loop counter
67 C952 CB1B      play2: rr      e      ; data -> cy flag
68 C954 160F      ld      d,15   ; volume := 8
69 C956 3802      jr      c,play3  ; if data = 0 then
70 C958 1600      ld      d,0      ; volume := 0
71 C95A 3E08      play3: ld      a,8
72 C95C CD00C0    call    sound
73 C95F CD6CC9    call    delay
74 C962 10EE      djnz    play2
75                ;
76 C964 DD23      inc      ix
77 C966 2B        dec      hl
78 C967 7D        ld      a,l
79 C968 B4        or      h      ; end ?
80 C969 20DE      jr      nz,play1 ; no , continue
81 C96B C9        ret
82                ;
83 C96C 3E05      delay: ld      a,5      ; delay counter
84 C96E 3D        delay1: dec     a
85 C96F C26EC9    jp      nz,delay1
86 C972 C9        ret

```



# 第4章 Stellarコンパイラ

---

4-1 Stellarコンパイラの概要

4-2 エディタ

4-3 モニタ

4-4 コンパイラ

---

Stellarコンパイラ・システムはエディタ、コンパイラ、ローダー、モニタが一体となっており、これらはオンメモリで動作するので高速です。また、豊富なライブラリを用意しているので、リアルタイム・ゲームやシステム・プログラムなどの開発に役立ててください。



## 4-1

# Stellarコンパイラの概要

Stellar コンパイラ・システムの特徴は次のとおりです。

- ① エディタは文字列の検索、置換、ブロックの編集（移動、コピーなど）が可能で、CP/Mのワード・スターのようなフルスクリーン・エディタです。
- ② Stellar コンパイラ（4-4 参照）は、オンメモリのため高速です。オブジェクトはGVRAM（グラフィック RAM）に出力し、ローダーによりメイン・メモリにロードします（このため、X 1 には、GVRAM が必要です）。
- ③ モニタは、HuBASIC のモニタの機能にブレーク・ポイントの設定、レジスタの表示・変更、さらに、16 進数の加減乗除などの機能を追加しています。
- ④ BASIC にある APSS, FILES の機能がカセット・テープに対して使用できます。
- ⑤ X 1 の広いメモリ空間を利用し、ソース・プログラムのエリアは約 30 Kバイトほどあります。

## 4-1-1 コマンドの説明

X 1 の Stellar は次の 10 個のコマンドを用意しています（各コマンドの最後には $\square$ キーを押すこと）。コマンドはスクリーン・エディットでき、大文字、小文字のどちらでもよく混在可能です。

### ① SED

プログラムを作成するために、エディタを起動します（4-2 で詳しく述べます）。

### ② COMP [/P]

エディタで作成した Stellar のプログラムをコンパイルします。オブジェクトは GVRAM に出力します。“/P”を付けるとコンパイル時の情報をプリンタにプリントします（4-



4で詳しく述べます)。

### ③ LDOBJ [/P]

GVRAMに出力されたオブジェクトをメモリにロードします。“/P”を付けるとロード時の情報をプリンタにプリントします(4-4で詳しく述べます)。

### ④ RUN [ & Hnnnn] (nnnnは16進数)

プログラムを実行します。“RUN & Hnnnn”はnnnn番地から実行します(4-4で詳しく述べます)。

### ⑤ CONT

% breakで実行を中断したプログラムを再開します(4-4で詳しく述べます)。

### ⑥ MON

モニタを起動します(4-3で詳しく述べます)。

### ⑦ APSS+

カセット・テープの頭出しをFAST方向にします。SHIFT+BREAKキー、^C (CTRLキーとCを同時に押す)、またはカセットのストップ・キーを押すことにより途中で実行を止めることができます。

### ⑧ APSS-

カセット・テープの頭出しをREW方向にします。SHIFT+BREAKキー、^C、またはカセットのストップ・キーを押すことにより途中で実行を止めることができます。

### ⑨ FILES [/P]

ファイルの一覧表を画面に表示します。“/P”を付けるとファイルの一覧表をプリンタにプリントします。

### ⑩ BOOT

IPL BOOTします(電源ONと同じです)。

## 4-1-2 ファンクション・キー

コマンド・モード時において、ファンクション・キーの設定は表4-1のようになっています(コールド・スタート時に設定)。




新たにファンクション・キーの設定を変えるにはコマンド・モードの時“MON [F6]”と打ちモニタに入り、ファンクション・キーバッファを変更してください。変更後、ホット・スタート（G 1 4 A 3 [F4]）すれば、変更したファンクション・キーが画面の下に表示されコマンド・モードになります。ファンクション・キーバッファは0 F 42 Hから始まります。1つのファンクション・キーバッファの長さは16バイトからなり、10個のエリア（合計で16×10=160バイト）を持っています。ファンクション・キーバッファの構造は、最初の1バイトが定義バイト数（0～15）、残りの15バイトが定義文字です。定義した文字の残りの部分にはスペース（20 H）を設定してください。

表4-1 コマンドモード時のファンクションの内容

F1	SED
F2	COMP
F3	LDOBJ
F4	CONT
F5	RUN
F6	MON
F7	APSS+
F8	APSS-
F9	FILES
F10	RUN&H



# 4-2 エディタ

コマンド・モードのときに F 1 を押すか、または“SED 

エディタの使用中にコマンドがわからなくなったときは ^ J (CTRL キーと J を同時に押す) を押すとコマンド表 (表 4-2) が表示されるので参考にしてください。コマンドは、

- ・カーソル移動
- ・インサート&デリート
- ・ファインド&リプレス
- ・ブロック・オペレーション
- ・その他

に大別できます。表中で、たとえば ^ Q F は CTRL キーを押しながら Q と F を続けて打つことを意味します。

ファンクション・キーの設定はエディタに入ると表 4-3 のように設定されます。

表4-2 エディタのコマンド表

●カーソル移動

^ S,←	文字左へ	^ D,→	文字右へ
^ A	語左へ	^ F	語右へ
^ E,↑	行上へ	^ X,↓	行下へ
^ R	ページ上へ	^ C	ページ下へ
^ W	スクロール・ダウン	^ Z	スクロール・アップ
^ QR,F8	テキストの先頭へ	^ QC,F9	テキストの最後へ
^ QB,F6	カーソルをブロックの先頭へ		
^ QK,F7	カーソルをブロックの最後へ		
^ I,HTAB	TAB	^ H,DEL	文字左へ

つづく



表4-2 つづき

●インサート & デリート	
^ V	挿入モードの切り換えへ
^ N	1行挿入
^ G	カーソルの下の1文字を削除
F 4	カーソルの前の1文字を削除
^ T	カーソルの右の1語削除
^ Y	カーソルのある行の削除
^ QY,F10	カーソルの位置から行末まで削除

●ファインド&リプレイス			
^ QF	検索	^ QA	置換
^ L	再検索		

●ブロック・オペレーション			
^ KB,F1	ブロックの始点マーク	^ KB,F2	ブロックの終点マーク
^ KC	ブロックのコピー	^ KV	ブロックの移動
^ KY	ブロックの削除	^ KH,F3	マーカの削除
^ KP	ブロックの印字	^ KS	プログラムのセーブ
^ KR	ブロックのリード	^ KW	ブロックのライト
^ KQ	プログラムの削除		

●その他	
^ J	ヘルプ・メッセージの表示
^ P	プログラムの印字
^ QL	ライン・サーチ
^ QM,F5	コマンド・モードに戻る

表4-3 エディタ使用時のファンクション・キーの内容

F1	^ KB	ブロックの始点マーク
F2	^ KK	ブロックの終点マーク
F3	^ KH	マーカの削除
F4	^ H ^ G	カーソルの前の1文字を削除
F5	^ QM	コマンド・モードに戻る
F6	^ QB	カーソルをブロックの先頭へ
F7	^ QK	カーソルをブロックの最後へ
F8	^ QR	カーソルをテキストの先頭へ
F9	^ QC	カーソルをテキストの最後へ
F10	^ QY	カーソルの位置から行末まで削除



## 4-2-1 カーソル移動

カーソルは BASIC を使っているときのカーソルの動きとは少々異なり、文字のないところにはカーソルを移動できません（多少違和感があると思いますが、すぐに慣れるでしょう）。プログラムを1行打ち込んだら␣キーを押しますが、␣キーの ASCII コード（0DH）も1文字として扱います。

カーソルを移動させるには、次のコマンドまたはキーを使います。

### ①カーソル・キー

BASIC と同様に上下に1行、左右に1文字単位で移動します。

### ②カーソル・キーと同じ動作をするもの

^Eで上、^Xで下、^Dで右、^Sで左へカーソルが移動します。これらのキーは CTRL キーの近くにあり、左手の小指で CTRL キーを押しながら人差し指や中指を使って片手で操作できます。

### ③スクロール

^Wで上、^Zで下へ1行分スクロールします。また、^Rで上、^Cで下へ画面の半分（12行分）をいちどにスクロールします。

### ④ワード単位のカーソル移動

^Fで1語分右へ、^Aで1語分左へカーソルが移動します。

### ⑤タブ

^I, HTAB キーでカーソルが8カラム移動します。これを使ってリストを見やすくできます。また、タブも1個の文字（ASCII コードの09H）として見なしています。

### ⑥その他

ファイル（この場合、ソース・プログラムのこと）の先頭へ移動するときは^QRまたはF8で、ファイルの終わりへは^QCまたはF9で移動します。また、^QBまたはF6でブロックの始め、^QKまたはF7でブロックの終わりに移動



します（ブロックはブロック・オペレーションを参照のこと）。

## 4-2-2 インサート&デリート

ここで説明するコマンドは文字や行の挿入や削除に使用します。

### ① オーバーライトとインサート・モードの切り換え

テキストの入力には2つのモード（オーバーライト・モードとインサート・モード）があり、最初にエディタを起動したときは、オーバーライト・モードになっています（画面の最上段に OVERWRITE と表示）。オーバーライト・モードでは、古いテキストを新しいテキストで置き換えたいときに使います。文字を入力するとカーソル位置にあった文字が入力した文字に置き換えられます。

インサート・モードは現在のテキストの中に新しいテキストを挿入していきます。新しいテキストを入力すると、カーソルより右にあるテキストは右に移動していきます（画面の最上段に INSERT と表示）。<sup>^</sup>Vで2つのモード間の切り換えを行ないます。

### ② 1行挿入

<sup>^</sup>Nで1行挿入ができ、カーソルの位置は変わりません。また、インサート・モードのときに<sup>␣</sup>キーを押せば1行挿入ができ、カーソルは次の行の先頭に移動します。これを利用すると行の分割ができます。分割するにはカーソルを行の分割したい所に移動して、<sup>^</sup>Nまたは<sup>␣</sup>キーを押します。

### ③ 1文字削除

<sup>^</sup>Gでカーソルの下の1文字を削除します。F4でカーソルの前の文字を削除します（F4は<sup>^</sup>H<sup>^</sup>Gと同様）。<sup>^</sup>Gを利用して行の連結ができます。連結するには行の最後にリターン・コード（0 DH）が入っていることを利用して、これを<sup>^</sup>Gで削除すると次の行と連結されます。

### ④ 1行削除



^ Yでカーソルのある行を削除します。

#### ⑤カーソル以後の削除

^ Tでカーソルのある位置から右をワード単位で削除します。

^ QY または F 15でカーソルのある位置から行の終わりまで削除します。^ Y, ^ Tと混同しないようにしてください。

## 4-2-3 ファインド&リプレイス

これはソース・プログラムの中から任意の文字列を検索（ファインド）したり，他の文字列に置き換える（リプレイス）コマンドです。

### ①ファインド

^ Q Fを押すと画面の左上で“Find?”と聞いてくるので，搜したい（50文字までの）文字列を入れ $\square$ キーを押します。このときの文字列の訂正はDELキーを使います（カーソル・キーは使えません）。

検索文字列を指定すると，次に“Options? (? For Info)”と聞いてきます。以下のオプション（大文字，小文字共に可）が使えます（オプションの並び方は自由）。

B…逆方向に検索します。つまり，現在のカーソルの位置からファイルの先頭に向かって検索を行ないます。

n…nは1から99までの数です。現在のカーソルの位置から数えて，n番目に表われる検索文字列を検索します。

?…オプション・メニューを表示します。この状態で再びオプションの指定ができます。

オプションのデフォルトは順方向検索（ファイルの最後に向かう検索），n = 1です。



#### オプションの例

B …現在のカーソルの位置から逆方向に1番目に表われる検索文字列を検索します。

10 …10番目に表われる検索文字列を検索します。

10B

b 10…この例は2つとも同じ機能で、現在のカーソルの位置から逆方向に10番目に表われる検索文字列を検索します。

オプションの並びを入れたら、最後に $\square$ キーを押してください。オプションで何も指定しないで $\square$ キーを押すと、 $n=1$ のオプションを指定したのと同じです。また、オプションで“B”のみを指定すると“B 1”を指定したことになります。検索文字列が見つかり、カーソルはその文字列のところに移動します。検索は $\wedge L$ を使って繰り返すことができます。

#### ②リプレイス

$\wedge QA$ を押すとファインドと同様に“Find?”と聞いてくるので検索文字列を入力します。次に“Replace With?”と聞いてくるので置き換えたい(50文字までの)文字列(置換文字列)を入力します。ここで、単に $\square$ キーを押せば検索文字列は空文字列に置き換えられる(つまり削除される)ことになります。さらに、“Options? (? For Info)”と聞いてくるので、以下のオプション(大文字、小文字共に可)を使います(オプションの並び方は自由)。

B…逆方向に検索置換します。つまり、現在のカーソルの位置からファイルの先頭にむかって検索置換を行ないます。

G…大域検索置換をします。つまり、現在のカーソルの位置にかかわらずテキスト全体を探索置換します。



n...n は 1 から 99 までの数です。現在のカーソルの位置から数え始めて n 回検索文字列を置換します。

N...確認なしで置き換えます。検索文字列を見つけるごとに止まって “Replace (Y/N) ?” と聞くことはありません。

?...オプション・メニューを表示します。この状態で再びオプションの指定ができます。


オプションのデフォルトは順方向に n = 1 で確認あり（検索文字列を見つけるごとに止まって, “Replace (Y/N) ?” と聞く）検索置換です。

また、検索は SHIFT+BREAK キーでやめることができます。

#### オプションの例

N 5...検索文字列を挿して確認なしで置き換えることを 5 回繰り返します。

G N...検索文字列を挿して確認なしで置き換えることをテキスト全体にわたって行ないます。

オプションの並びを入れたら最後に  キーを押してください。

### ③再検索&再検索置換

^ L で最後に実行した検索あるいは、検索置換を再実行します。このコマンドにより、同じ文字列を検索するときなどいちいち文字列を指定する必要がなくなります。これは新しく検索、検索置換するまで有効です。

## 4-2-4 ブロック・オペレーション

ブロック・オペレーションは、ファイル（テキスト）の一部を移動、コピー、削除などをしたり、ブロックの印字、カセット・テープへブロックのセーブなどに使われます。

ブロックをマークするには、テキストの中のブロックにし



たい部分の先頭の文字にカーソルを移動し、「ブロックの始め」のマークを、最後の文字の後に「ブロックの終わり」のマークを付けます。このようにマークをすればブロックを移動したり、コピーしたりできます。

#### ①ブロックの始点をマーク

^KBまたはF1でブロックの始点をマークします。マーカは“→”で示されます。すでに始点のマークが指定されているときに新たに始点のマークを指定すると、前に指定したマーカは消され新しいマーカが設定されます。

#### ②ブロックの終点をマーク

^KKまたはF2でブロックの終点をマークします。マーカは“←”で示されます。すでに終点のマークが指定されているときに新たに終点のマークを指定すると、前に指定したマーカは消され新しいマーカが設定されます。

#### ③ブロックのコピー

^KCでマークしたブロックを現在のカーソルから先にコピーします。元のブロックは変化しません。マーカは元のブロックの方に付いています。カーソルの位置がブロック内にあるときはコピーされず、エラーメッセージも出ません。また、ブロックの指定が不完全のときはエラーメッセージが出ます。

#### ④ブロックの移動

^KVでマークしたブロックを元の位置から現在のカーソルの位置に移動します。元の位置からブロックは消え、マーカは新しい位置に移ったブロックに付きます。カーソルの位置がブロック内にあるときは移動できず、エラーメッセージも出ません。

#### ⑤ブロックの削除

^KYでマークしたブロックを削除します。マーカはブロックと共に削除され、カーソルがブロック内にあっても削除されます。

#### ⑥マーカの削除

^KHまたはF3で始点、終点マーカを消します。マー



カーは<sup>^</sup>Gや<sup>^</sup>Yで消すこともできます。マーカーを付けたまま、コンパイルするとエラーになるので必ず消してから行なってください。

#### ⑦ブロックの印字

<sup>^</sup>K Pでマークしたブロックをプリンタに印字します。

#### ⑧プログラムのセーブ

<sup>^</sup>K Sを押すと“Save Filename? :”と聞いてくるのでファイル名（ファイル名：13文字，拡張子：3文字）を入力して，最後に $\square$ キーを押してください。ファイル名の訂正はDELキー，カーソル・キーが使えます。ファイル名の指定をするとテキスト（ソース・プログラム）をすべてカセット・テープにセーブします。カセット・テープがセットされていないときや，テープ・エンドのときは何もしません。

セーブはASCII形式で行なわるので，BASICなどでASCIIセーブされたテキストと互換性があります。

#### ⑨プログラムのリード

<sup>^</sup>K Rを押すと“Read Filename? :”と聞いてくるので，ファイル名を入力して，最後に $\square$ キーを押してください。ファイル名の指定をすると，カセット・テープからリードしたテキストは現在のカーソルから先に挿入されます。ファイル名の指定をしなければ最初のファイルをリードします。ファイルはASCIIセーブされたものでなければリードできません。

#### ⑩ブロックのセーブ

<sup>^</sup>K Wを押すと“Write Filename? :”と聞いてくるので，ファイル名を入力して，最後に $\square$ キーを押してください。ブロックのマーカーはセーブされません。

#### ⑪プログラムの削除

<sup>^</sup>K Qを押すと“Abandon a File? (Y/N)”と聞いてくるので，Y（またはy）を押すとプログラムをすべて削除します。Y以外のキーを入力すると何もしません。



## 4-2-5 その他

### ①ヘルプ

^ Jで表 4-2 のようなコマンド表が表示されます。これによりコマンドを忘れたときは、マニュアルを見直す必要はありません（もともと ^ J というコマンドを忘れたときは論外ですが…）。

### ②プログラムの印字

^ Pでプログラム（テキスト）をすべてプリンタに印字します。

### ③ライン・サーチ


^ Q Lを押すと“Line Number? :”と聞いてくるので、カーソルを移動したい行番号（5桁までの数字）を入力して、最後に $\square$ キーを押します。このときの行番号の訂正はDELキーを使います（カーソル・キーは使えません）。指定した行番号が存在すればその行の先頭にカーソルが移動します。存在しない“0行”を入力するとテキストの先頭にカーソルは移動し、テキストの最後の行数よりも大きい番号を指定するとテキストの最後にカーソルが移動します。

### ④コマンド・モードに戻る


^ QMまたは、F 5でコマンド・モードに戻ります。



## 4-3 モニタ

“MON ”または F 6 でモニタが起動します。HuBASIC の内蔵のモニタとほとんど同じですが R, G コマンドが異なり X, H コマンドが追加されています。

### 4-3-1 R コマンド


このコマンドを実行するとモニタから抜け、コマンド・モードに戻ります。戻らないときは、スタックが破壊されているときですから、G コマンドで“G 14 A 3 ”としてホット・スタートしてください。

### 4-3-2 G コマンド

指定した番地のサブルーチンをコールします。このときにブレーク・ポイントを最大 2 個まで設定可能です。G コマンドはブレーク・ポイントの設定の仕方で、次の 3 種類の型があります。

① G n 

② G n    a 

③ G n    a    b 

(ここで n, a, b は 16 進数の 4 桁までの数値)

①は n で示された番地から始まるサブルーチンをコールするもので、n を省略した場合、現在のプログラム・カウンタ (X コマンド参照) が示すアドレスからのプログラムをコールします。

②はブレーク・ポイントを 1 個設定したもので、a で示される番地に達したときプログラムの実行がストップし、その



ときのレジスタとフラグを表示します。a の位置にある命令は実行されません。n を省略した場合、現在のプログラム・カウンタが示すアドレスからのプログラムをコールして、ブレーク・ポイントをa に設定します。

③はa とb という2 個のブレーク・ポイントを設定するもので、どちらかのブレーク・ポイントに達したときプログラムはストップします。n を省略した場合、現在のプログラム・カウンタが示すアドレスからのプログラムをコールして、ブレーク・ポイントをa とb に設定します。

## 4-3-3 Xコマンド

このコマンドでレジスタ、フラグの内容を表示・変更することができます。X コマンドには次の2 種類の型があります。

①X $\square$

②Xレジスタ名

①はレジスタ、フラグの内容を表示するものです。フラグは2 進数でも表示されます。ただし、裏レジスタの内容は表示しません。

②は指定されたレジスタの内容を変更するもので、指定できるレジスタはA, F, B, C, D, E, H, L, AF, BC, DE, HL, IX, IY, SP, PCです。

## 4-3-4 Hコマンド

このコマンドは16 進数の加減乗除を行ないます。H コマンドには次の4 種類あります。

①H a + b $\square$

②H a - b $\square$

③H a \* b $\square$



④  $H a / b$  

(a, bは16進数で最大4桁, H, a, 演算子, bの間には任意のスペースを置くことが可能, ただし同一行に限る)

①から③の演算結果は16進数で表示されます。④の演算結果は16進数で商, 余りの順序で表示されます。



## 4-4

## コンパイラ

Stellar は小規模プログラムの開発言語で、文法的には、C と Pascal を合わせたような構造（構造化プログラミング、全域変数と局所変数が可能、再帰的プログラミング）をしており、簡単なゲームや制御用のプログラムなどに適用できます。また、Stellar コンパイラのオブジェクト・プログラムは ROM 化可能です。

Stellar の最大の特徴は、変数のメモリ割りつけ、および演算がすべてバイト（符号なし 8 ビット）単位であることです。データがバイト長ですむようなプログラムなら、他のコンパイラに比べ小さなオブジェクト・プログラムを生成することができます。Stellar については『新言語作成の技法』（大貫広幸著、小社刊）で、CP/Mバージョンがアセンブル・リストで公開されています。

### 4-4-1 Stellarの構文と文法

構文は構文図（図 4-1）を参照してください。

ここでは、Stellar の文法を次の五つに分けて解説します。

- ・ Stellar プログラムの構成要素
- ・ プログラムの構造
- ・ 文
- ・ 式
- ・ 変数と定数

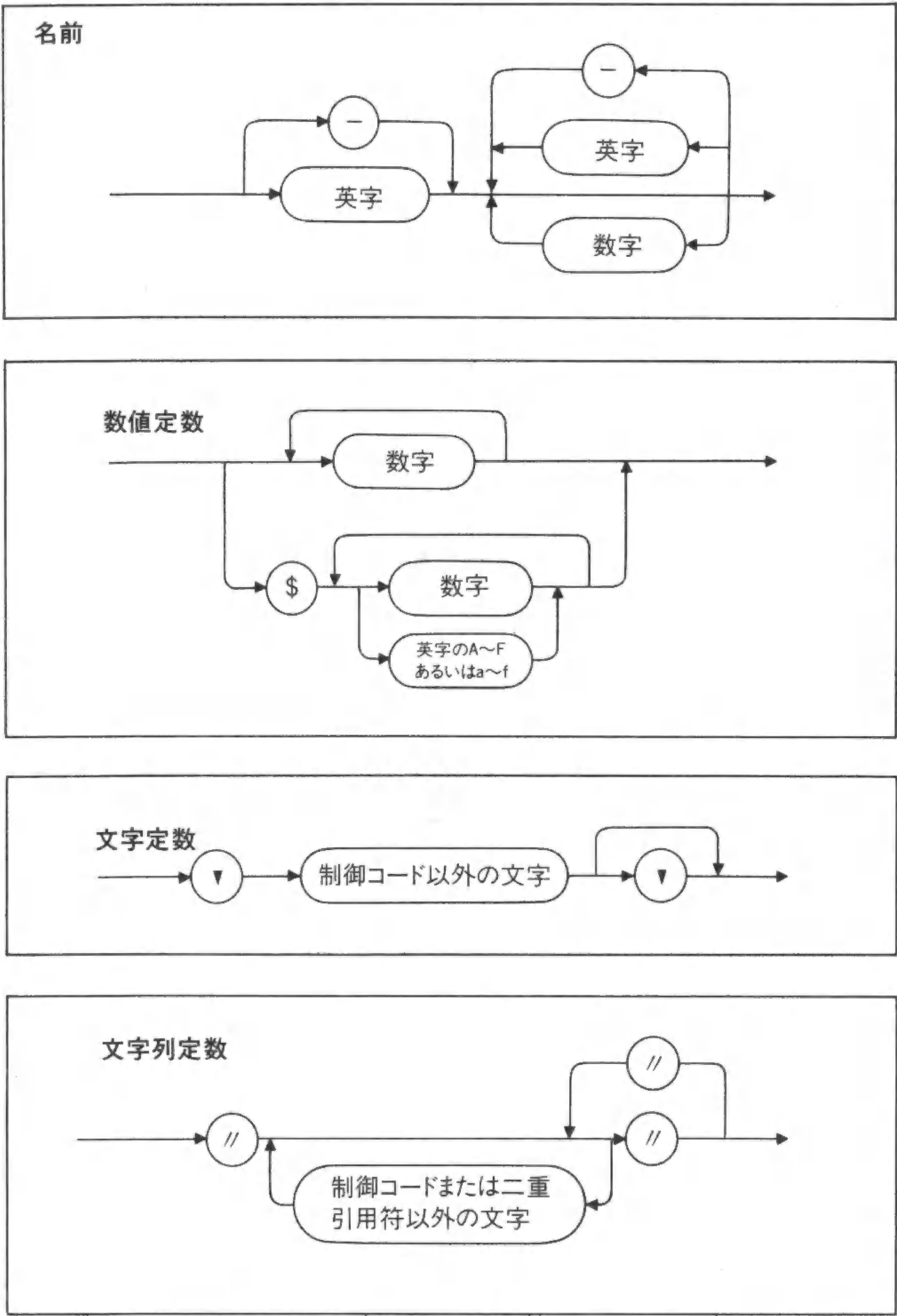
Stellar の言語的な特徴は以下の通りです。

- ① 英小文字ベースでプログラムを書く。
- ② サブルーチンはなく、すべて関数として記述する。
- ③ 関数は値を戻さなくてもよい。
- ④ 式の中で変数への代入ができる。また、代入をしない式もかける。



⑤ CPU 内のレジスタ (b, i x, i y) を使うことができる。

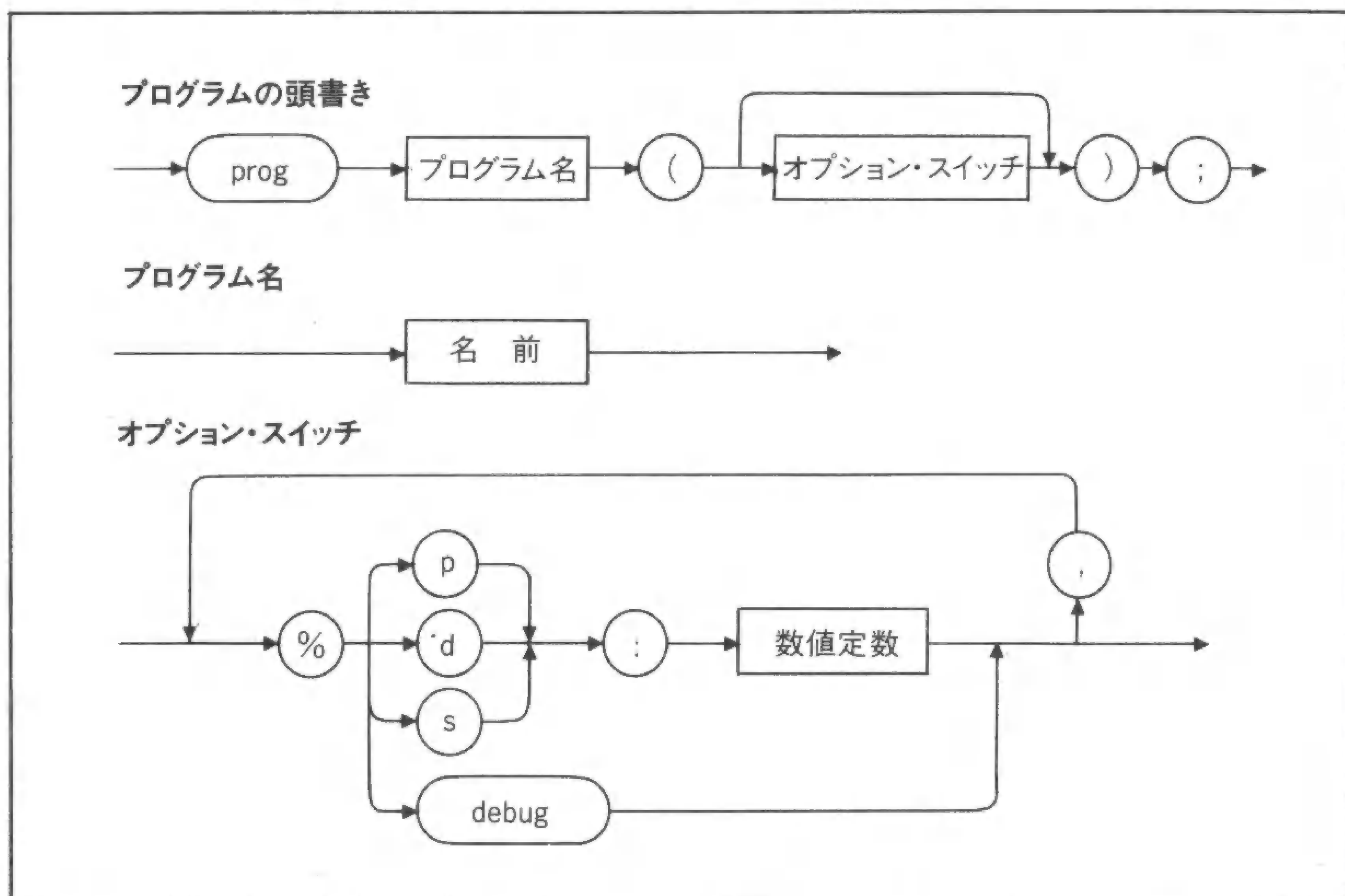
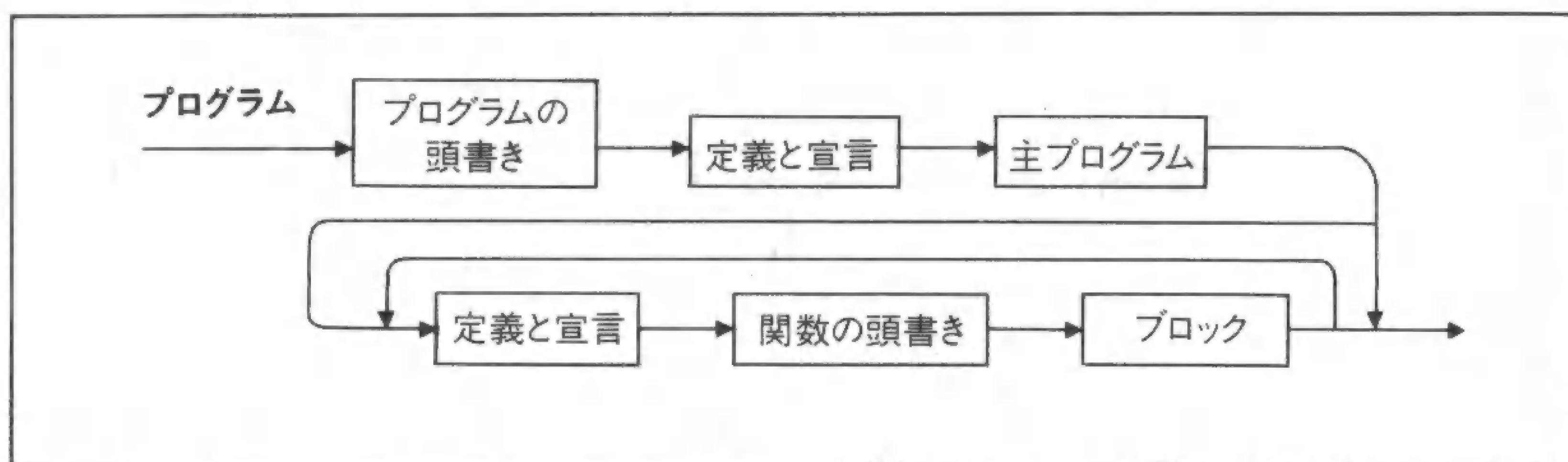
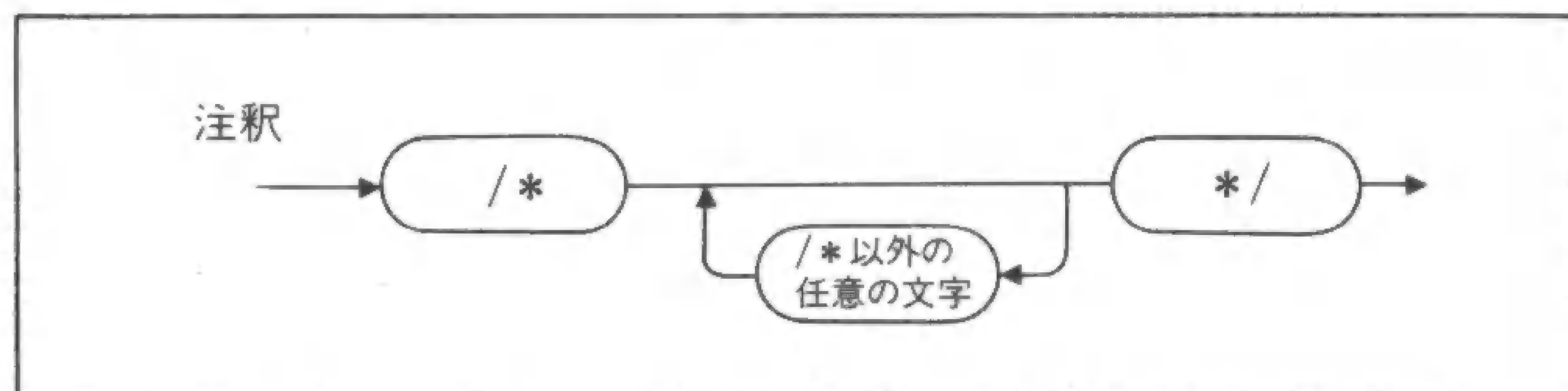
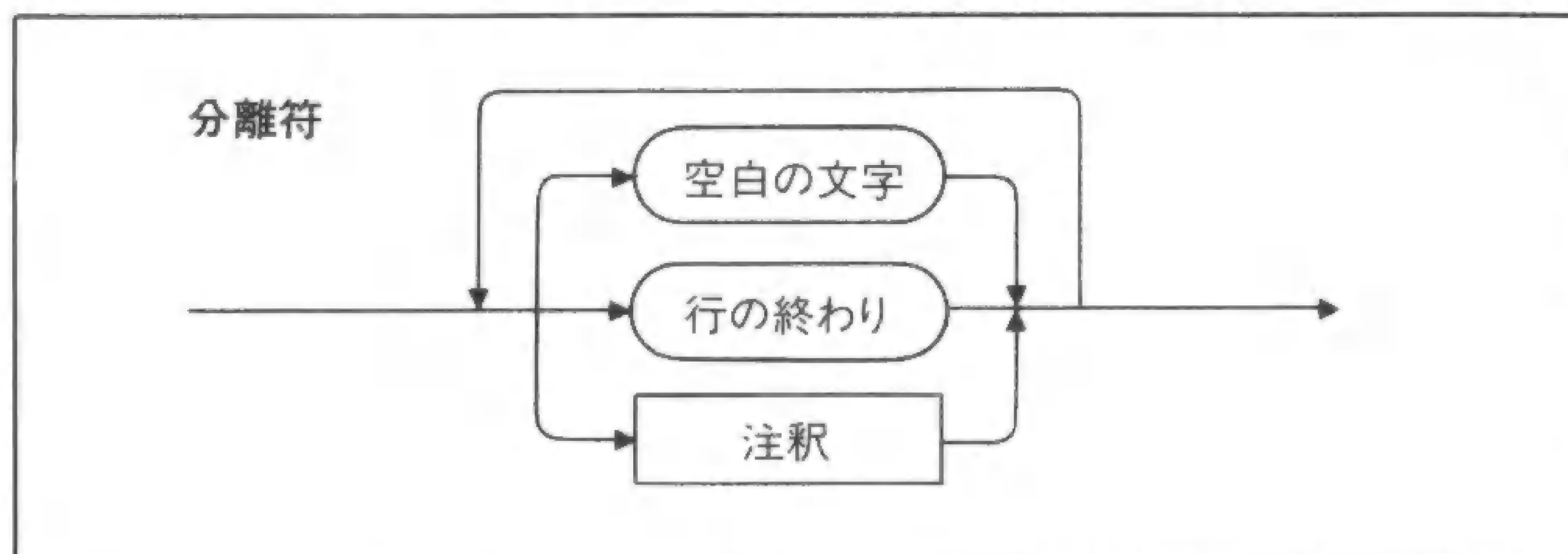
図4-1 Stellar構文図



つづく



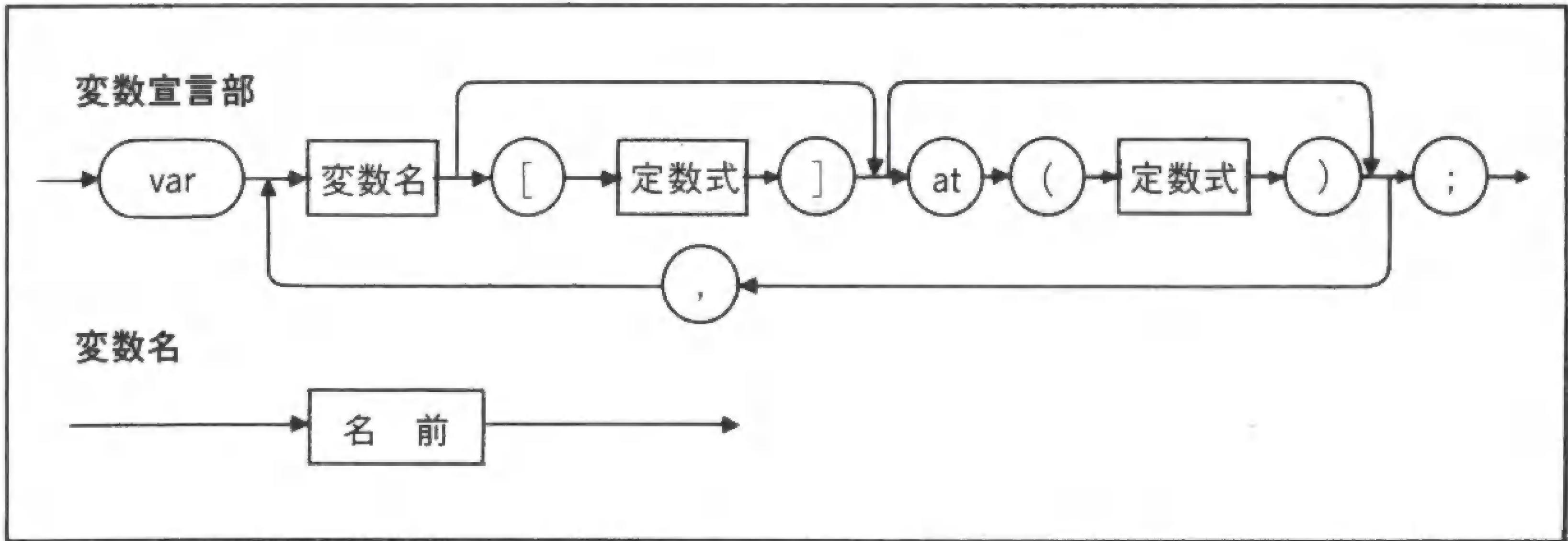
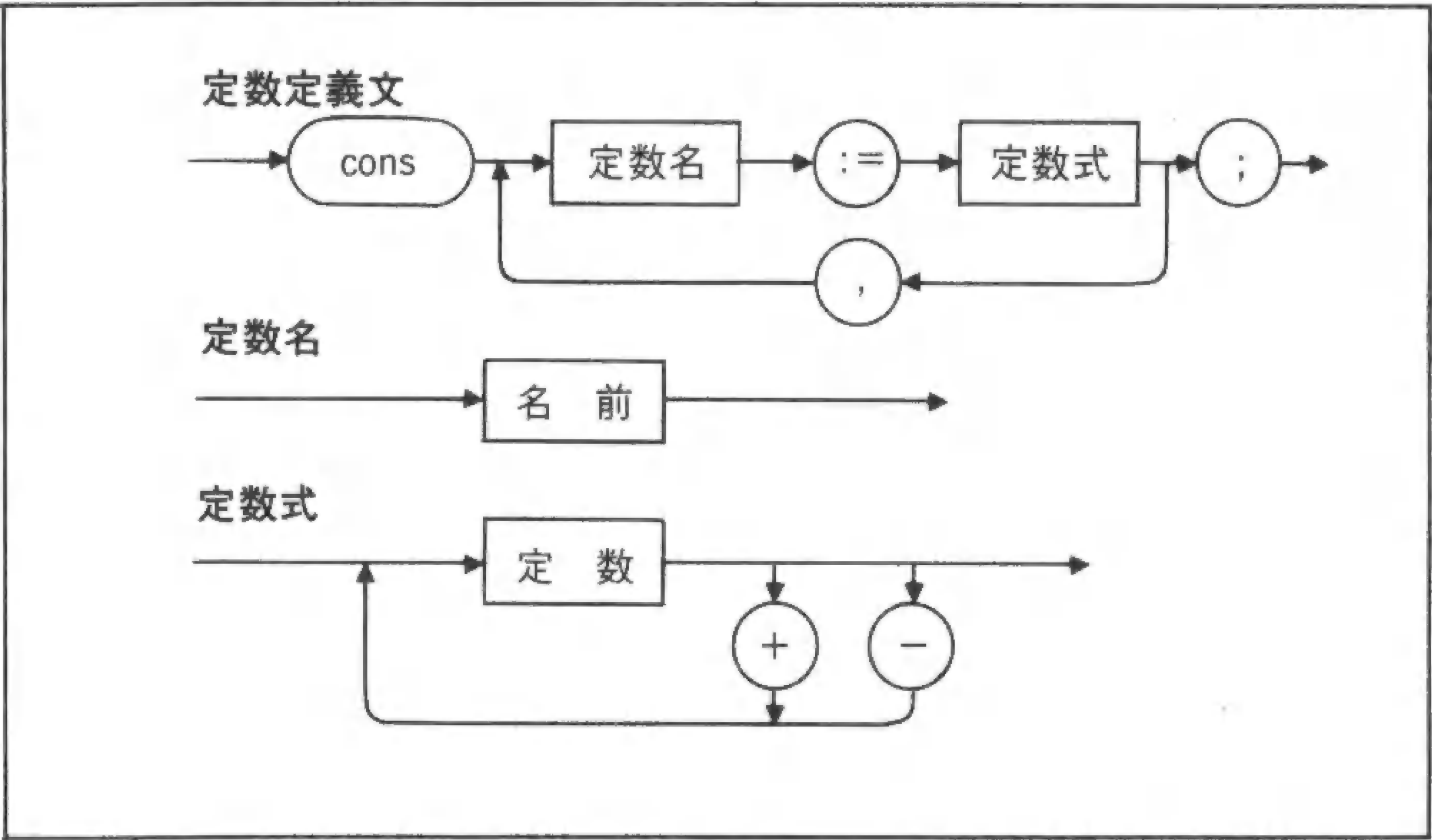
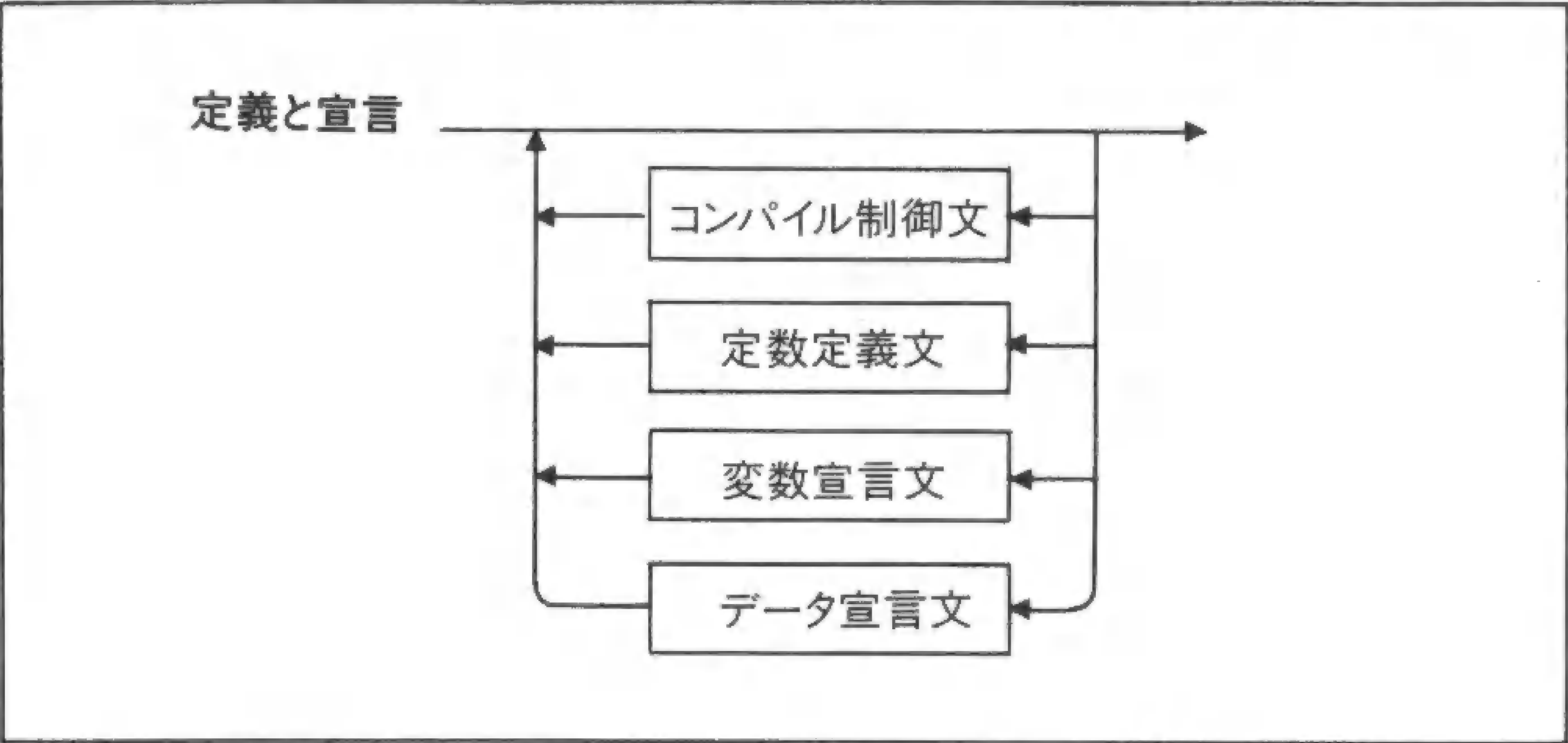
図4-1 つづき



つづく



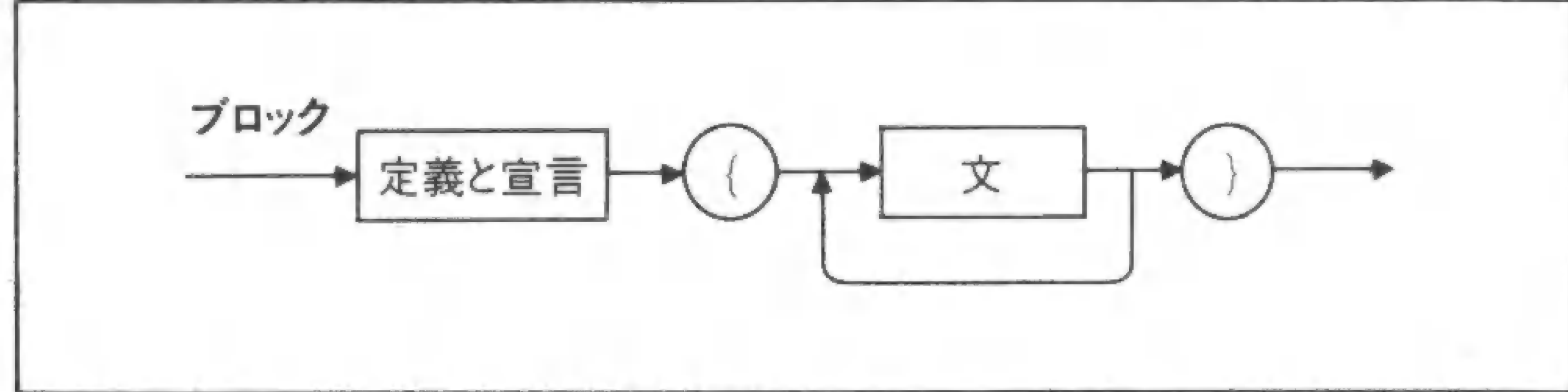
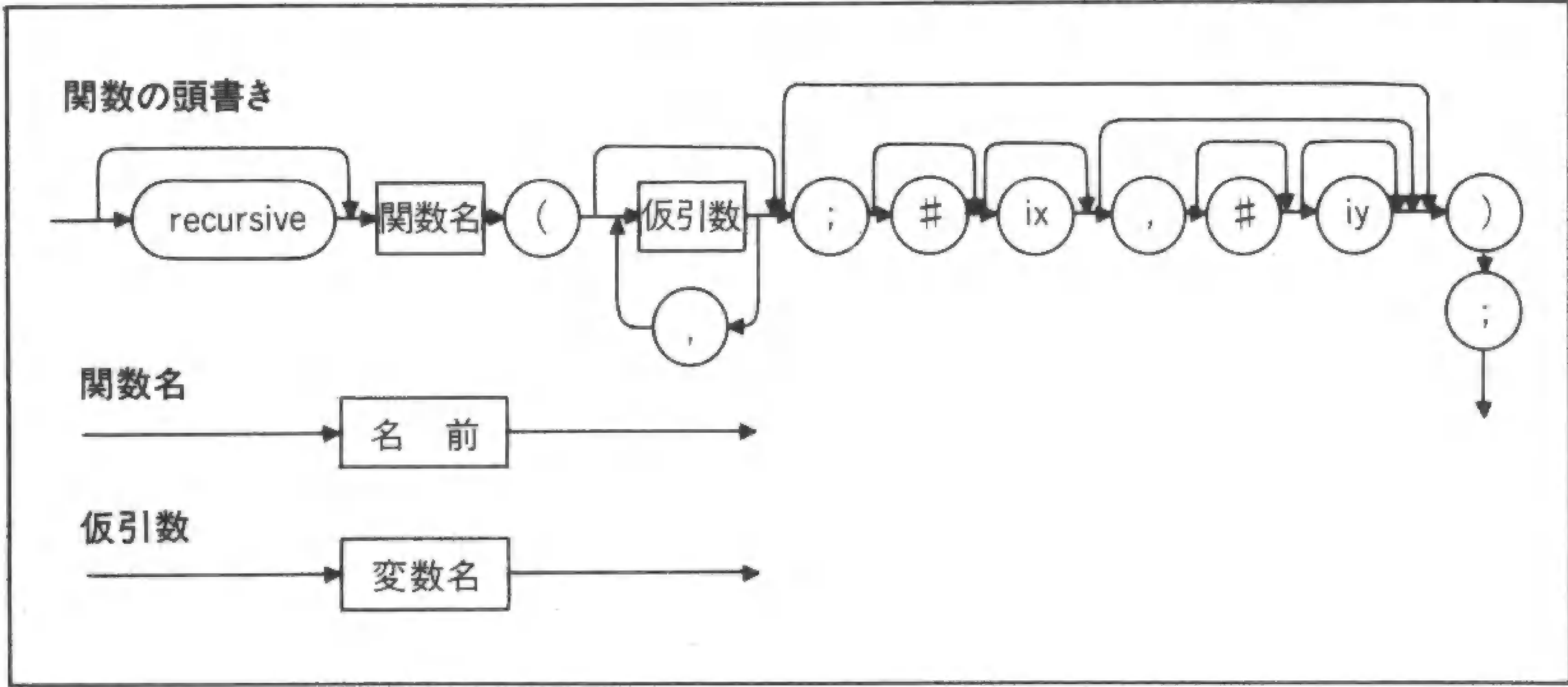
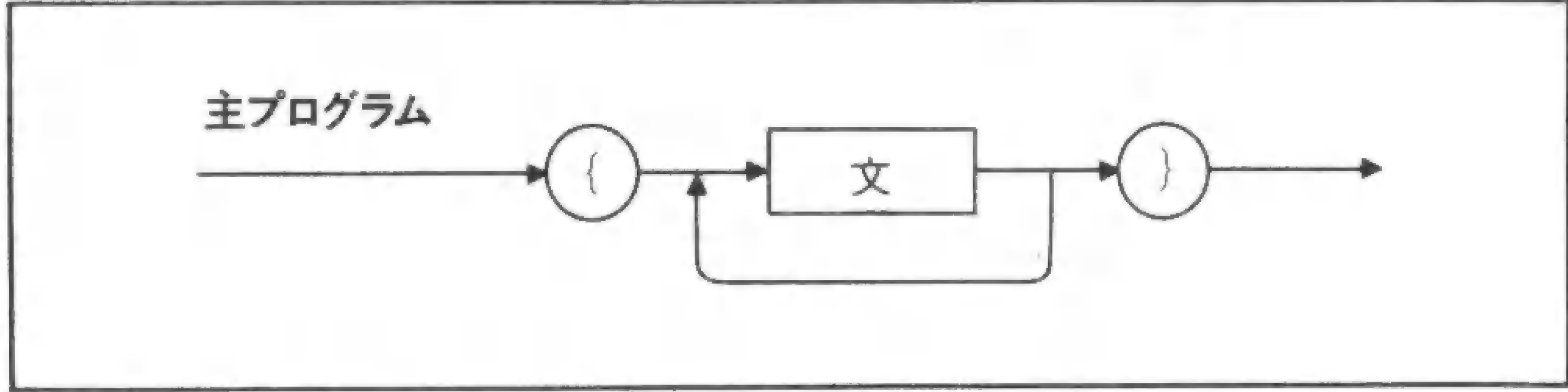
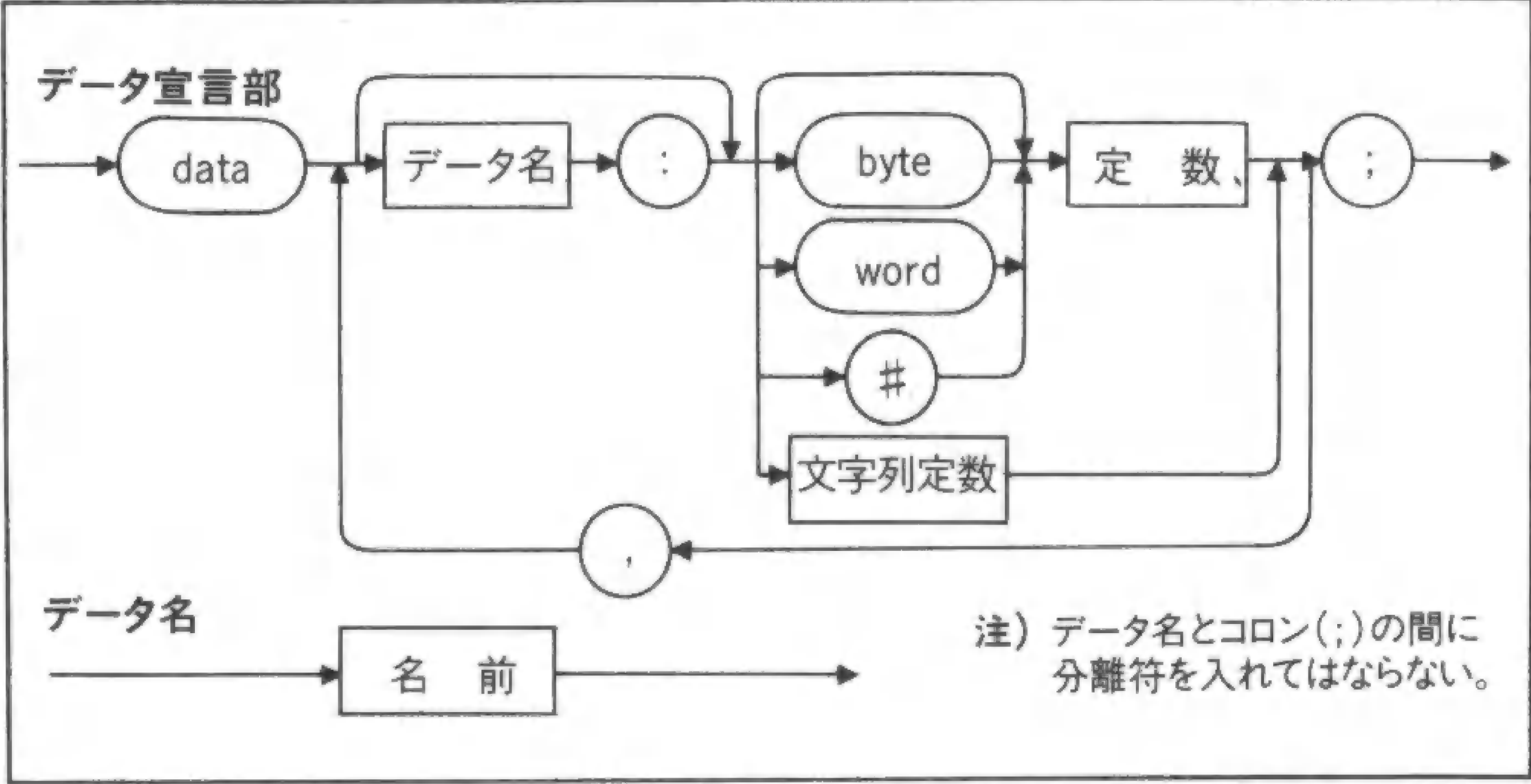
図4-1 つづき



つづく



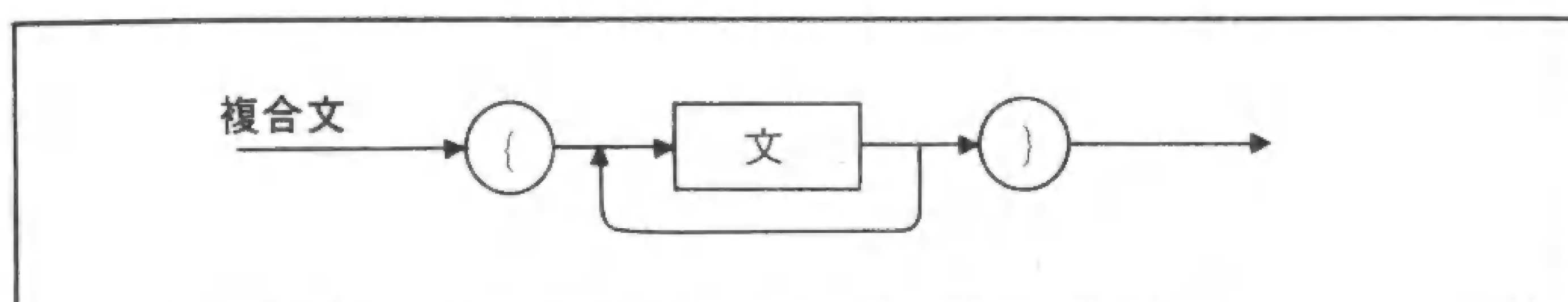
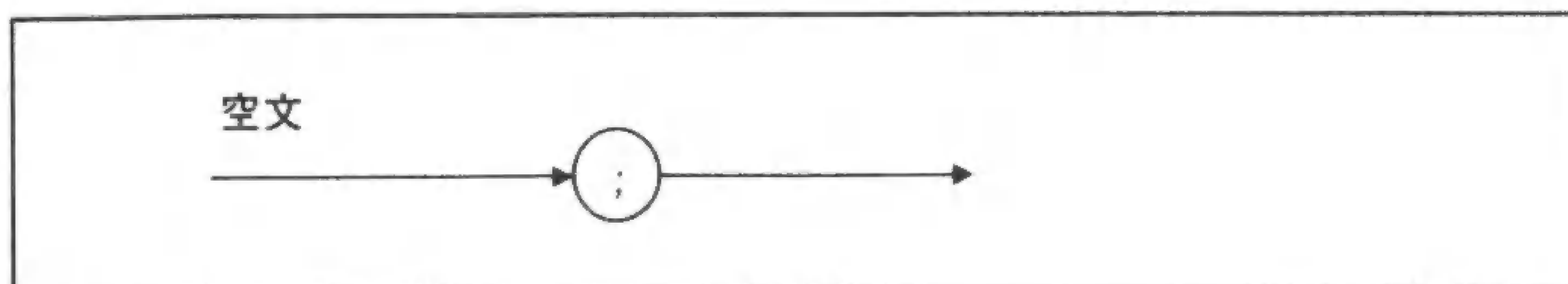
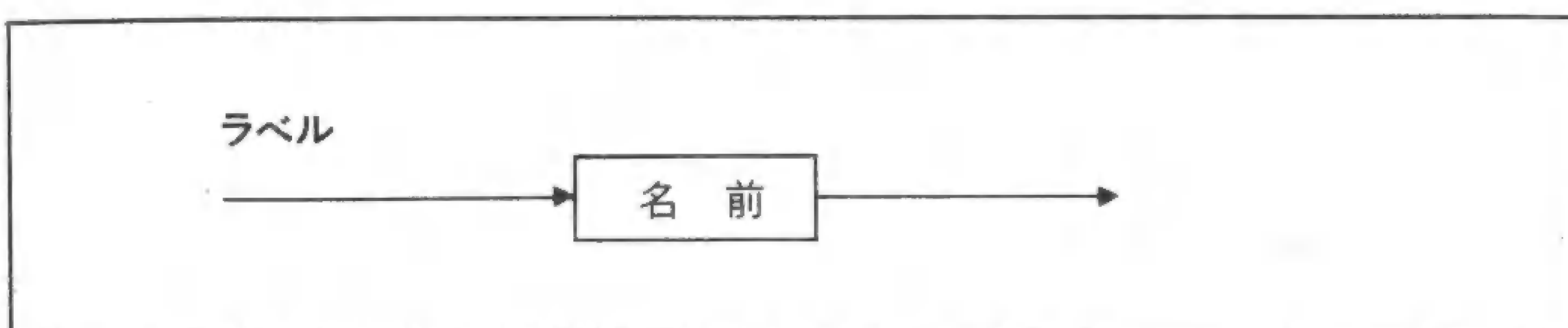
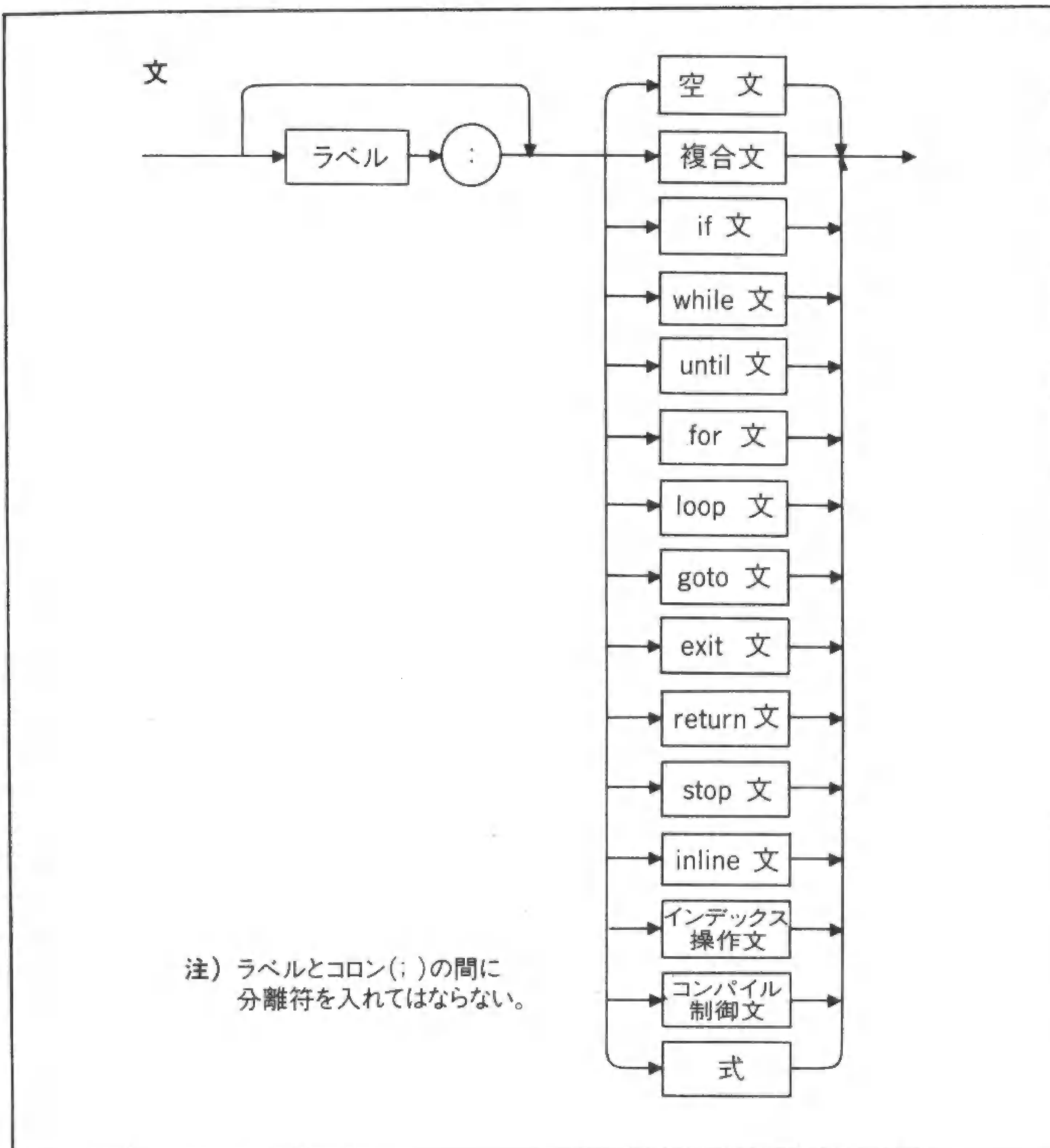
図4-1 つづき



つづく



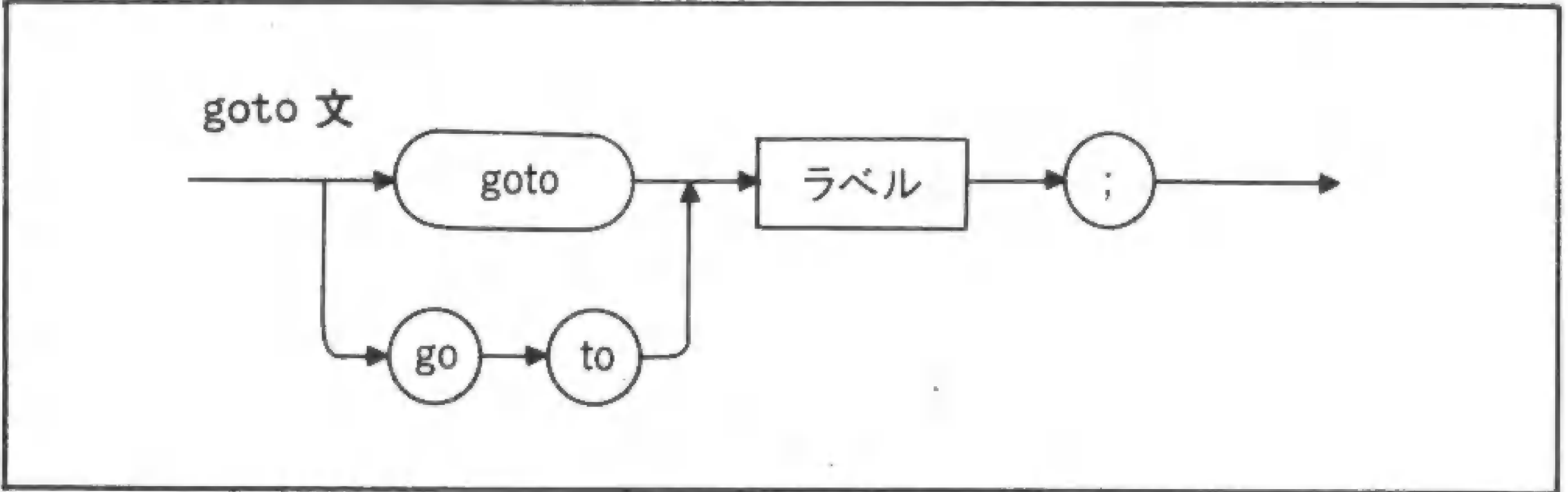
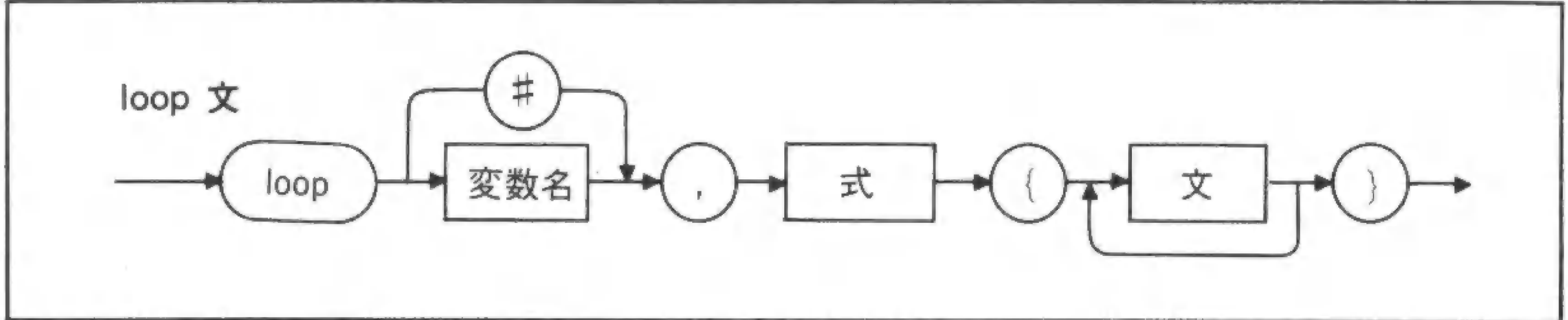
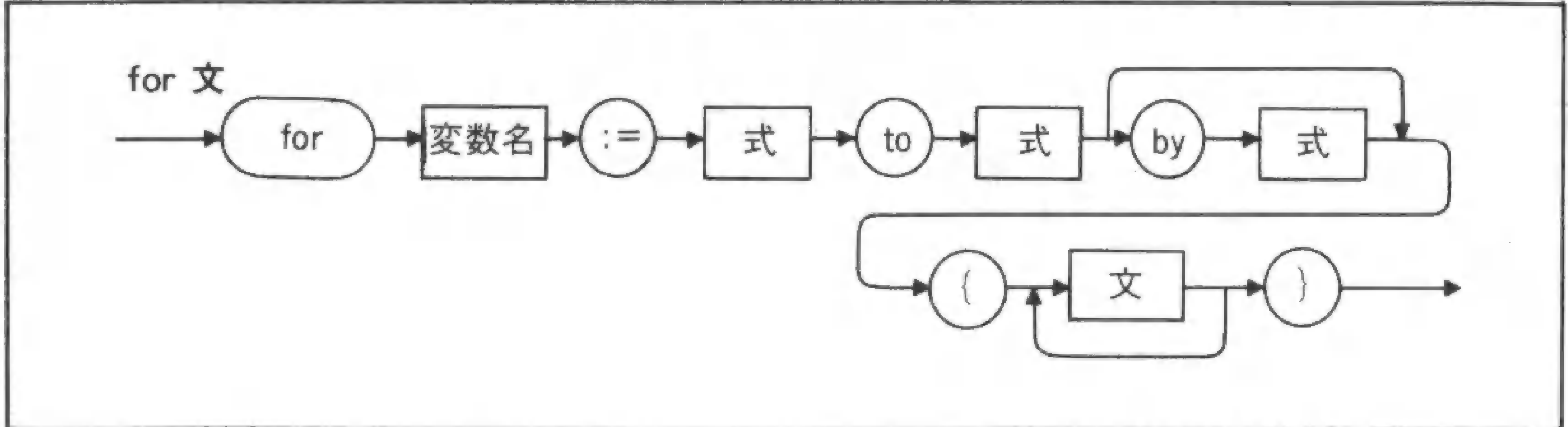
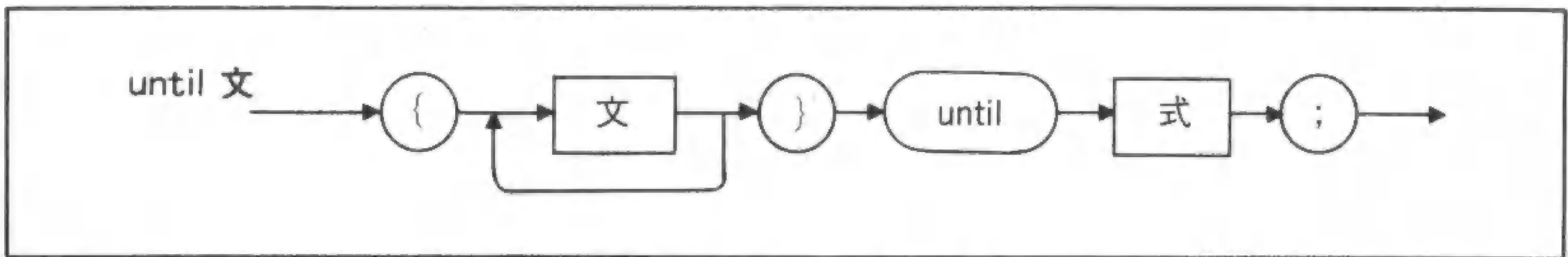
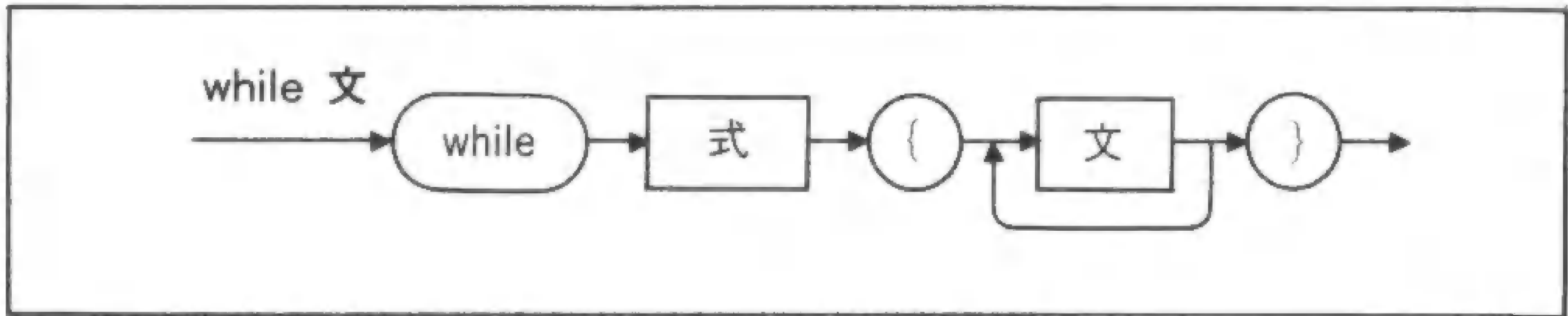
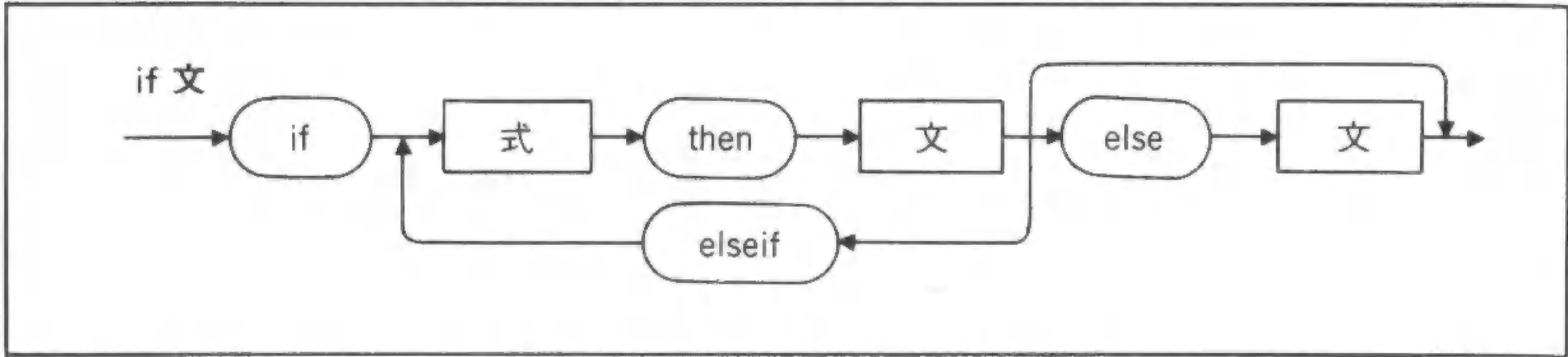
図4-1 つづき



つづく



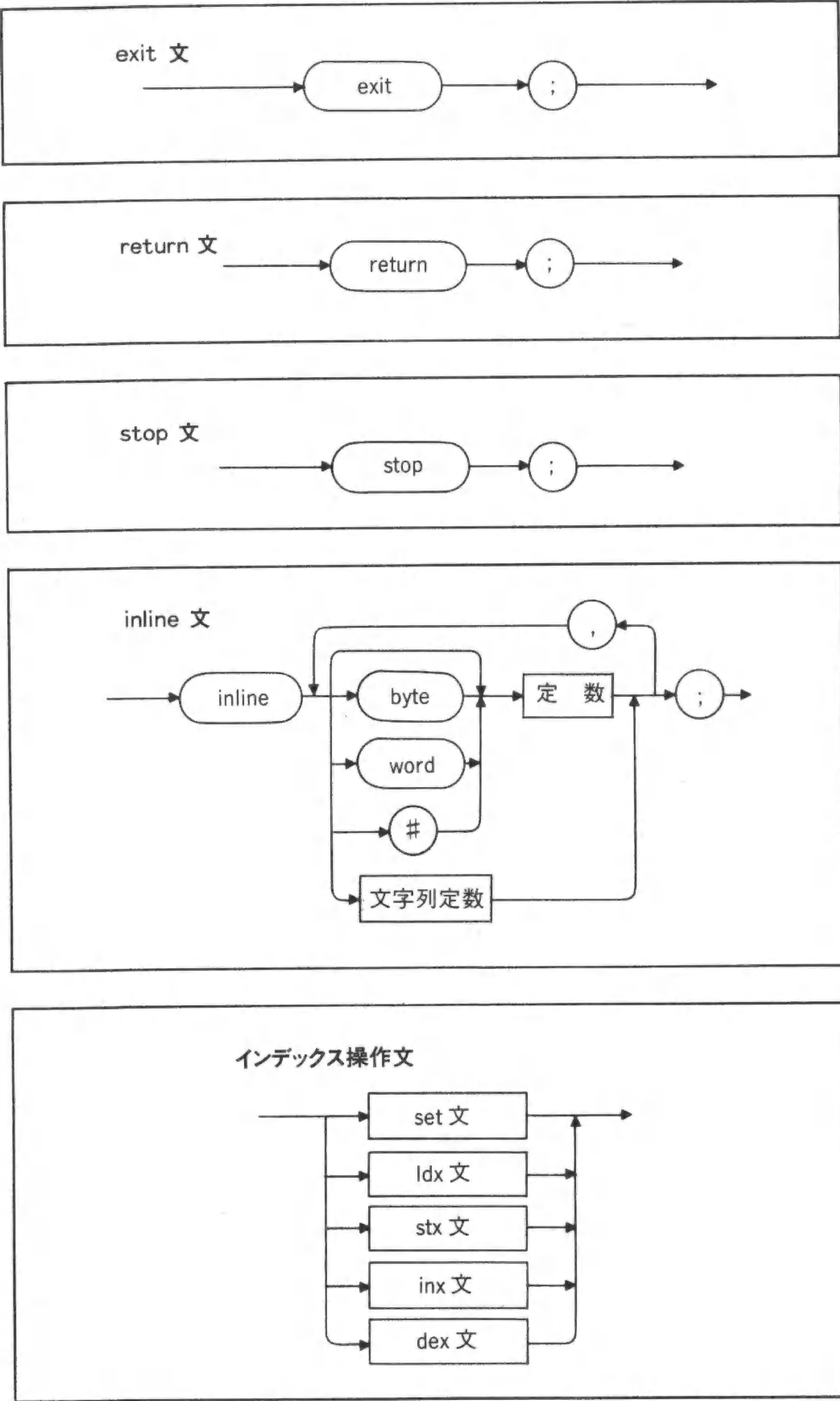
図4-1 つづき



つづく



図4-1 つづき



つづく



図4-1 つづき

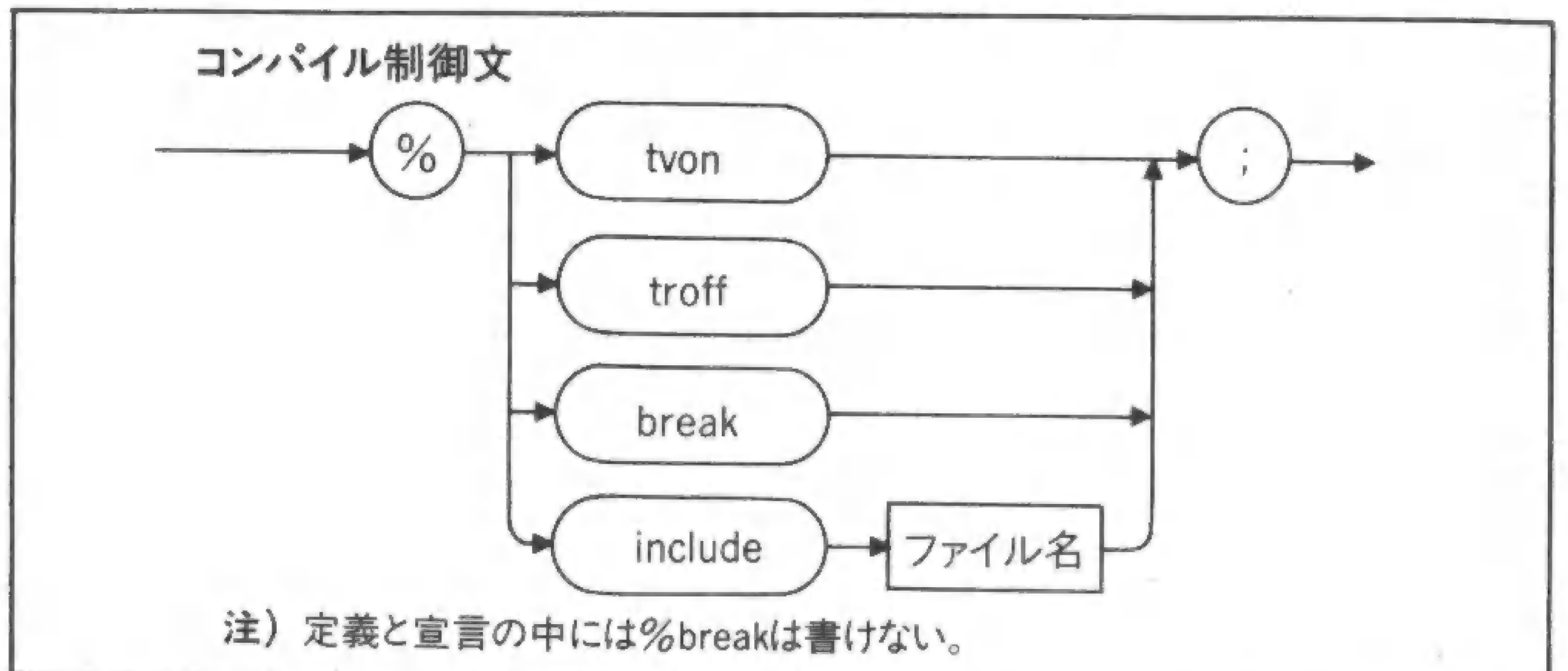
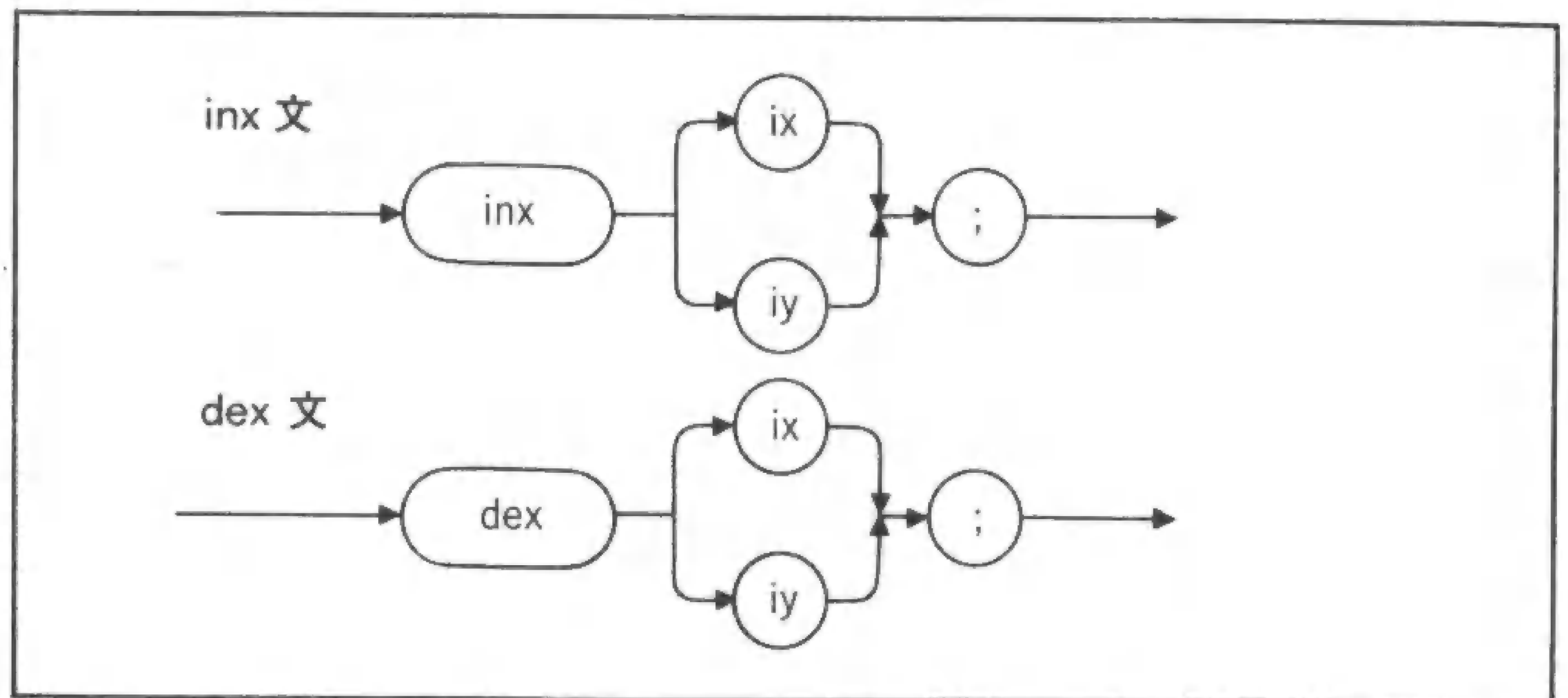
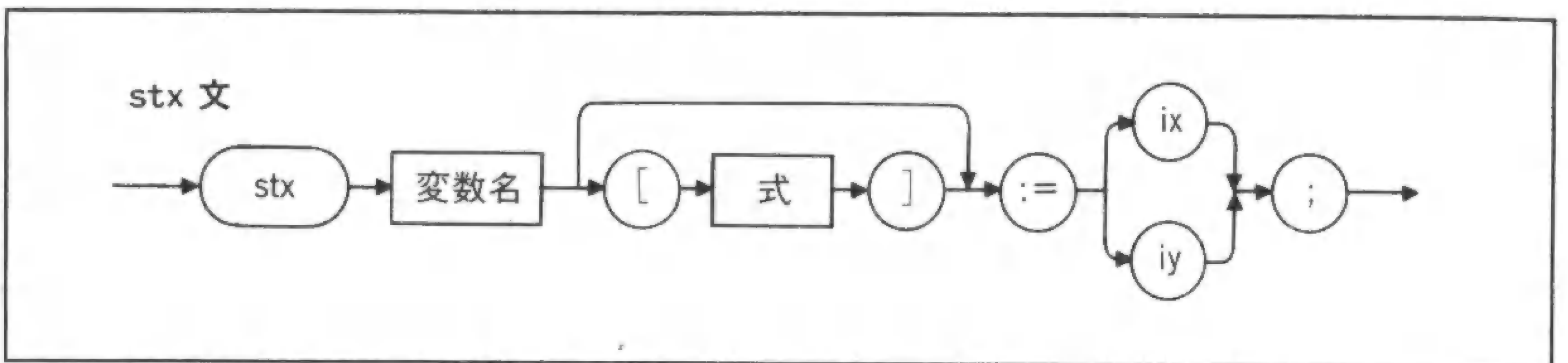
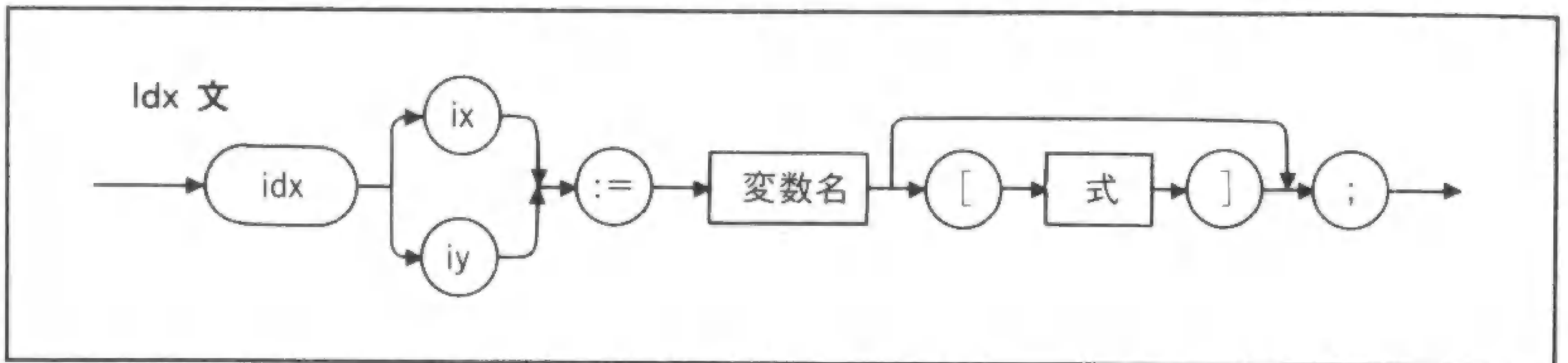
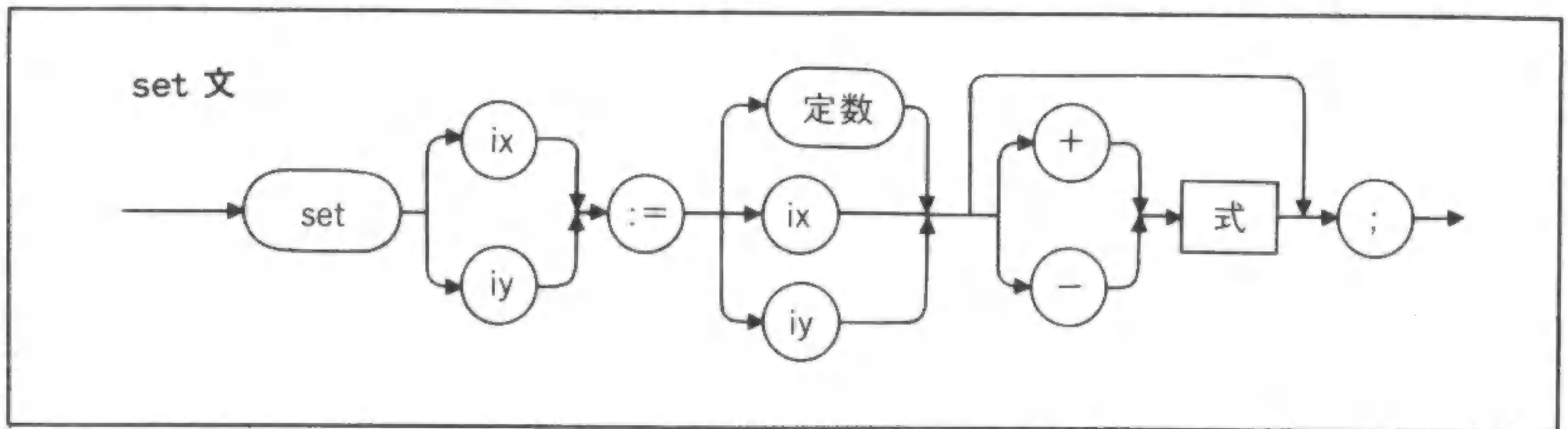
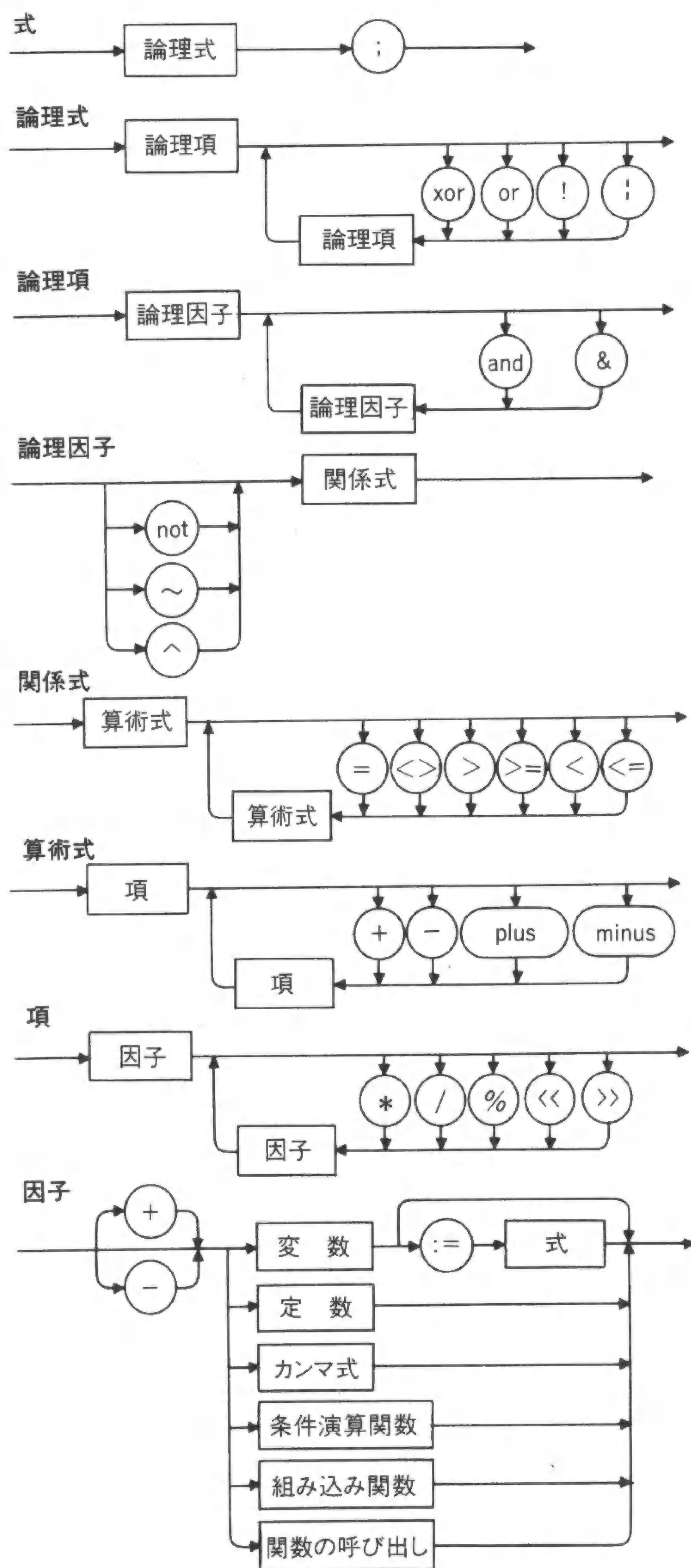




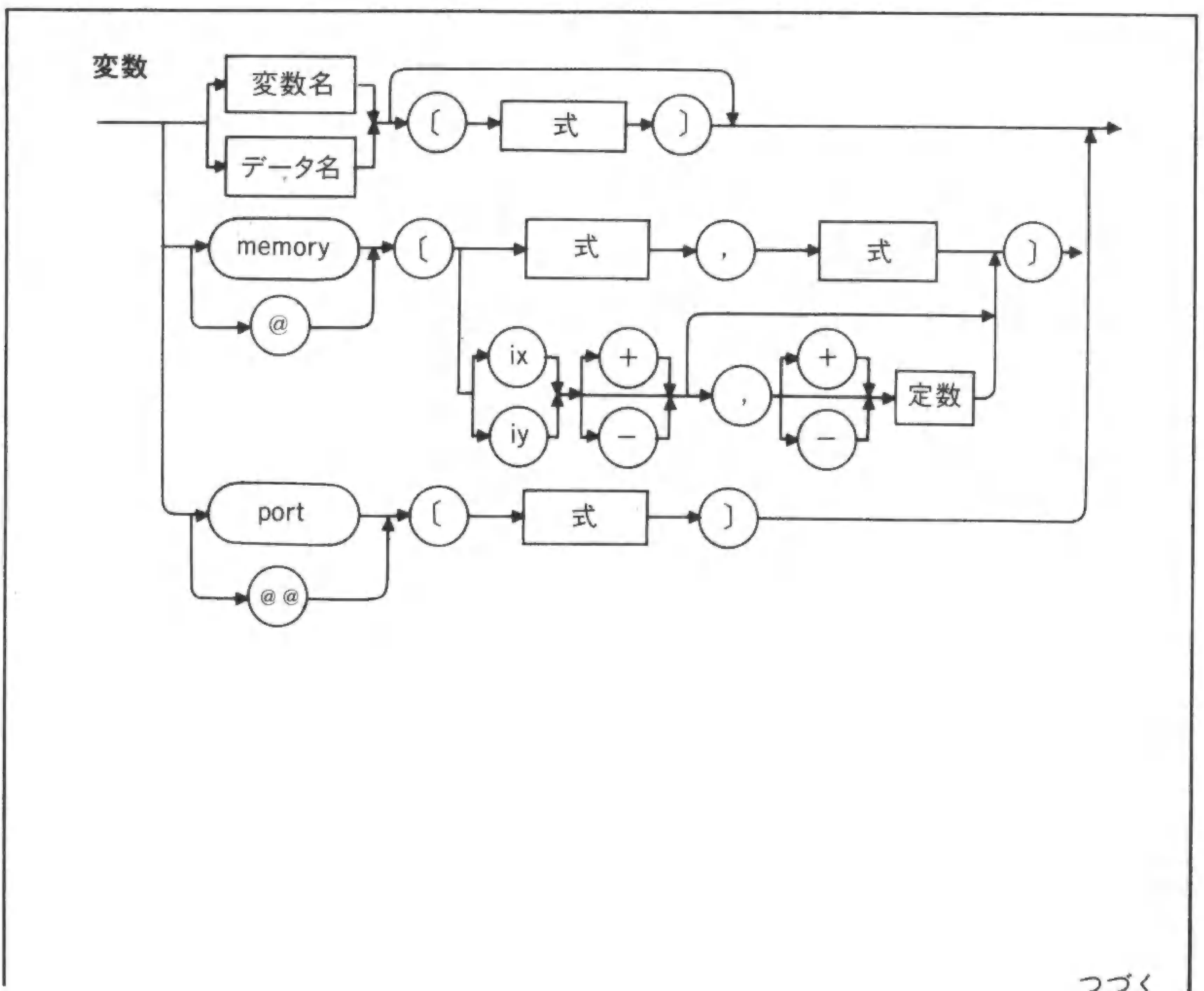
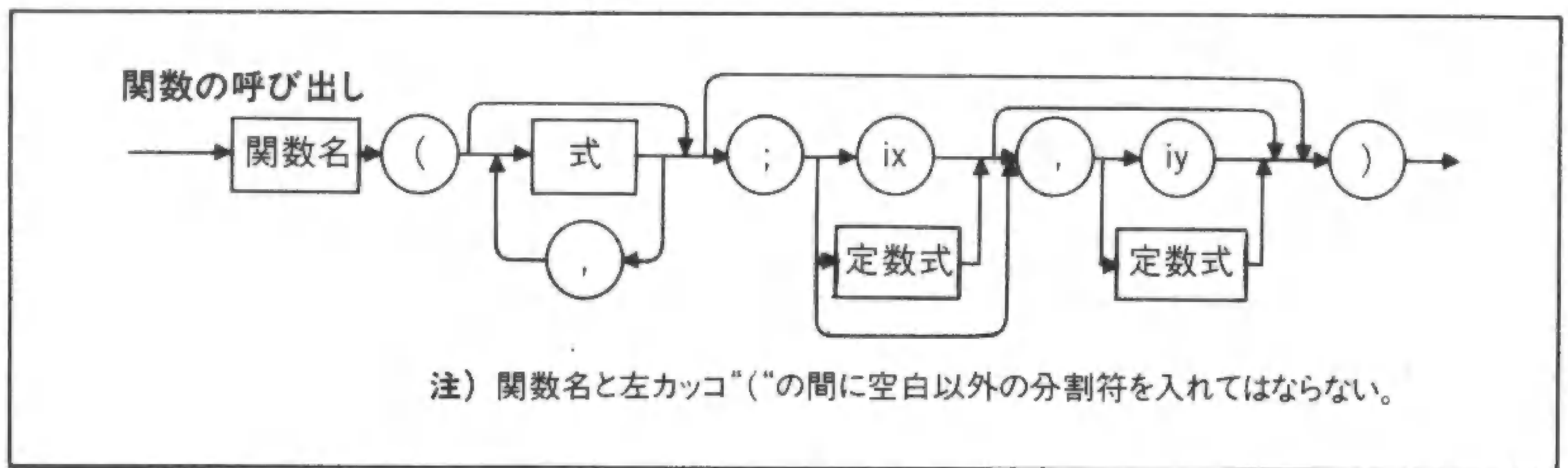
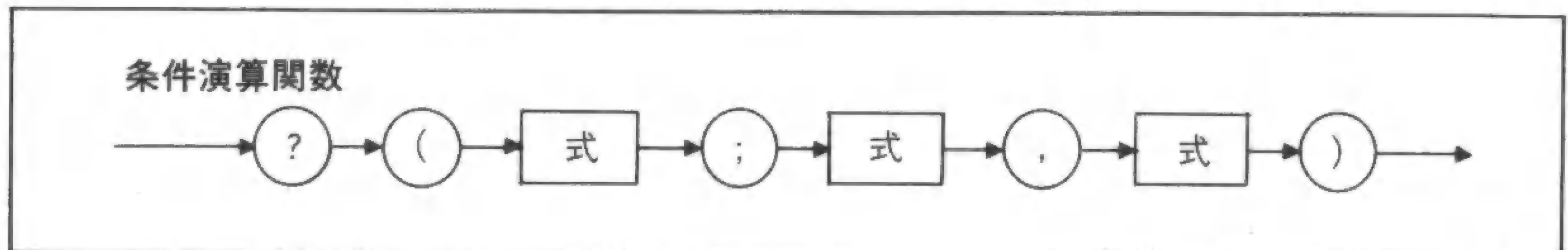
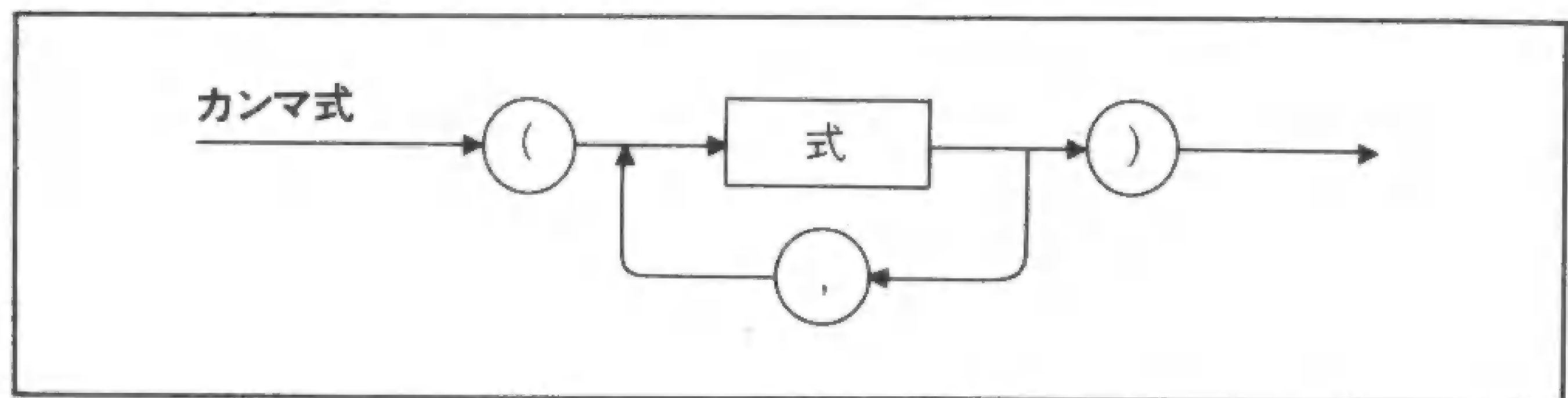
図4-1 つづき



つづく



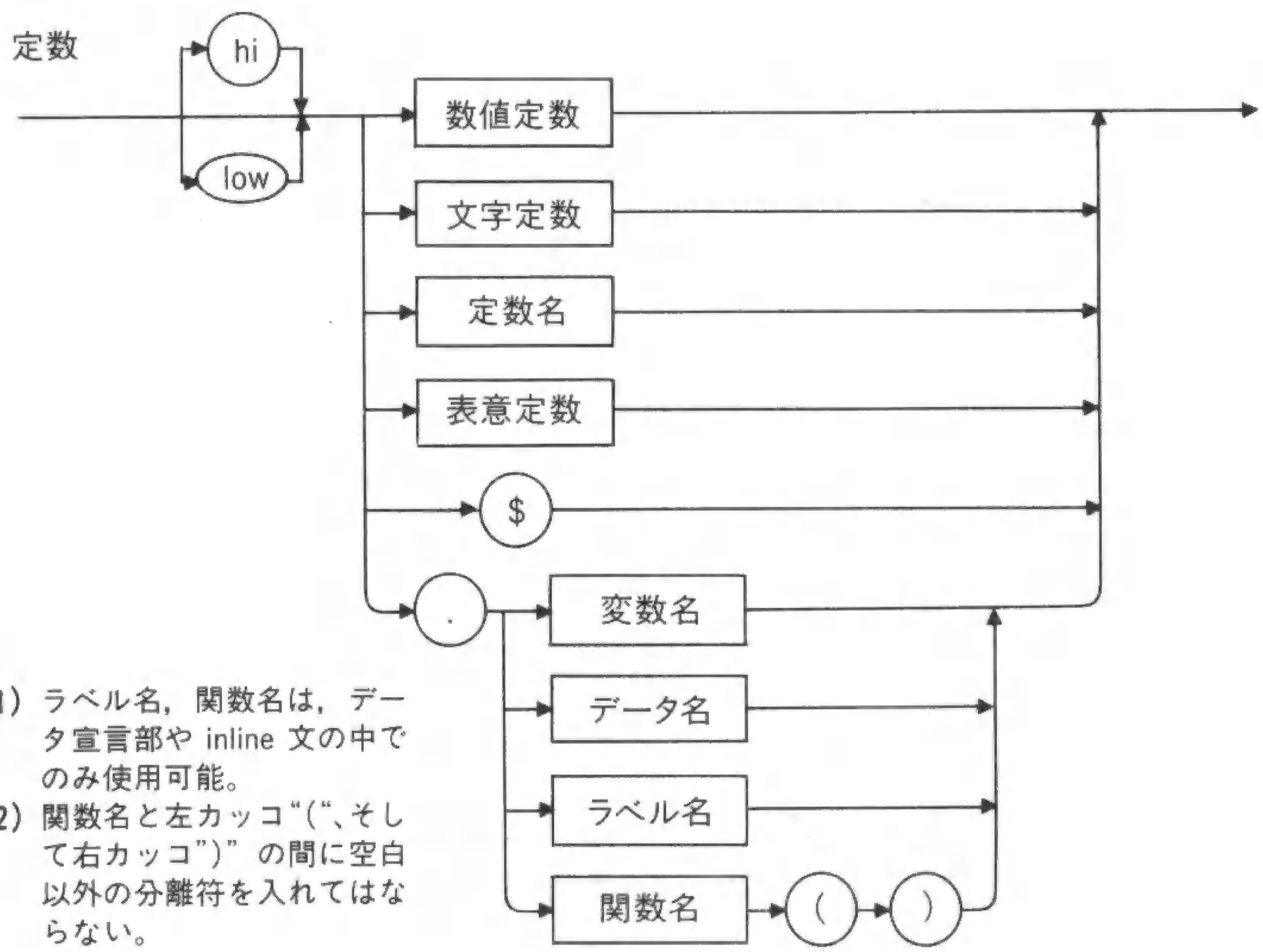
図4-1 つづき



つづく



図4-1 つづき



注1) ラベル名、関数名は、データ宣言部や inline 文の中でのみ使用可能。

注2) 関数名と左カッコ“(、そして右カッコ”)”の間に空白以外の分離符を入れてはならない。

[ 1 ] Stellar プログラムの構成要素

Stellar のプログラムは、予約語、名前、数値、定数、文字定数、文字列定数、特殊記号、それに分離符と注釈の八つで構成されています。

予約語

Stellar は 59 個の予約語を持っています (表4-4)。英大文字でも英小文字でも同じ意味になります。たとえば “WHILE”, “WHiLe”, “WhiLe” はいずれも予約語 “while” と判断されます。



表4-4 Stellarの予約語

and	dex	inc	minus	rl	to
at	debng	include	not	rlc	troff
break	else	inline	or	rr	tron
by	elseif	inx	overflow	rrc	until
byte	exit	ix	parity	set	var
carry	for	iy	plus	sign	while
cons	go	ldx	port	sra	word
data	goto	loop	prog	stop	xor
dec	hi	low	recursive	stx	zero
decj	if	memory	return	then	

### 名前

名前は変数や定数、関数などを表します。

名前は、英数字（A～Z，a～z，0～9）とアンダースコア（\_）からなる、数字以外の文字で始まる文字列です。長さに制限はありませんが、意味を持つのは最初の12文字までです。

#### ●正しい名前の例

ABC  
\_Ad0123  
a b c d e f g h i j k l m n…初めの12文字(1まで)  
までが名前となる。

a b \_XYZ

#### ●誤った名前の例

1AB…数字で始まっている。  
a b @ c d…名前の中に@（英数字でも \_ でもない）がある。  
low…予約語は名前として使えない。

### 数値定数

数値定数には、10進数と16進数があります。

10進数は数字（0～9）だけで構成された文字列で、0



～65535 までの値でなければなりません。16 進数はドル記号 (\$) で始まる数字と英字 A～F あるいは a～f で構成された文字列で、16 進数で 0～FFFF までの値でなければなりません。

例

```
10 進数  123  0043  1023  32924
          3292  255
16 進数  $ 0   $ 78  $ 002b  $ 3ff
          $ 80a6  $ ff
```

### 文字定数

文字定数は文字自身をデータとします。

文字定数は制御コード以外の 1 文字を引用符 (') で囲んだもので、文字コードを数値として扱います。右側の引用符は省略することができます。

例

```
'A'... $ 41 と同じ      'm'... $ 63 と同じ
'ア'... $ B1 と同じ     '"' ... $ 27 と同じ
'A ... $ 41 と同じ      '¥ ... $ 5 C と同じ
```

### 文字列定数

文字列定数は文字列自身をデータとしてメモリ上に格納する場合に使います。

文字列定数は二重引用符 (") で囲まれた 0 文字以上の文字列のことです。二重引用符を二つ続けて書けば文字列の中に一つの二重引用符を入れることができます。

例

```
"ABC012"      ...文字列ABC012がメモリに格納される。
"ABC""abc"    ...文字列ABC"abcがメモリに格納される。
""            ...空の文字列、何も格納されない。
"カナ 漢字"   ...カナ文字、漢字も使用できる。ただし
               漢字はシフトJISコードだけを使う。
```



## 特殊記号

特殊記号には 33 個の演算子と区切り記号があります (表 4-5)。

表4-5 特殊記号

!	(	,	:	<<	>	@	^	~
#	)	-	:=	<=	>=	@@	{	
%	*	.	;	<>	>>	[	!	
&	+	/	<	=	?	}		

## 分離符

分離符には空白文字 (タブも空白)、行の終わり、注釈の三つがあり、予約語、名前、定数、特殊記号を区切るために用いられます。これらが構文的に明確に識別できる場合には省略できます。

if    a>=\$b    then    b := a    and    \$F ;

↑

↑

↑

↑

↑

↑

↑

↑

↑

↑

↑

↑は分離符が必要な部分

↑は分離符が必ずしも必要でない部分

## 注釈

注釈は / \* で始まり \* / で終わる文字列です。 \* / の 2 文字をこの順で注釈の文字列中に書くことはできません。

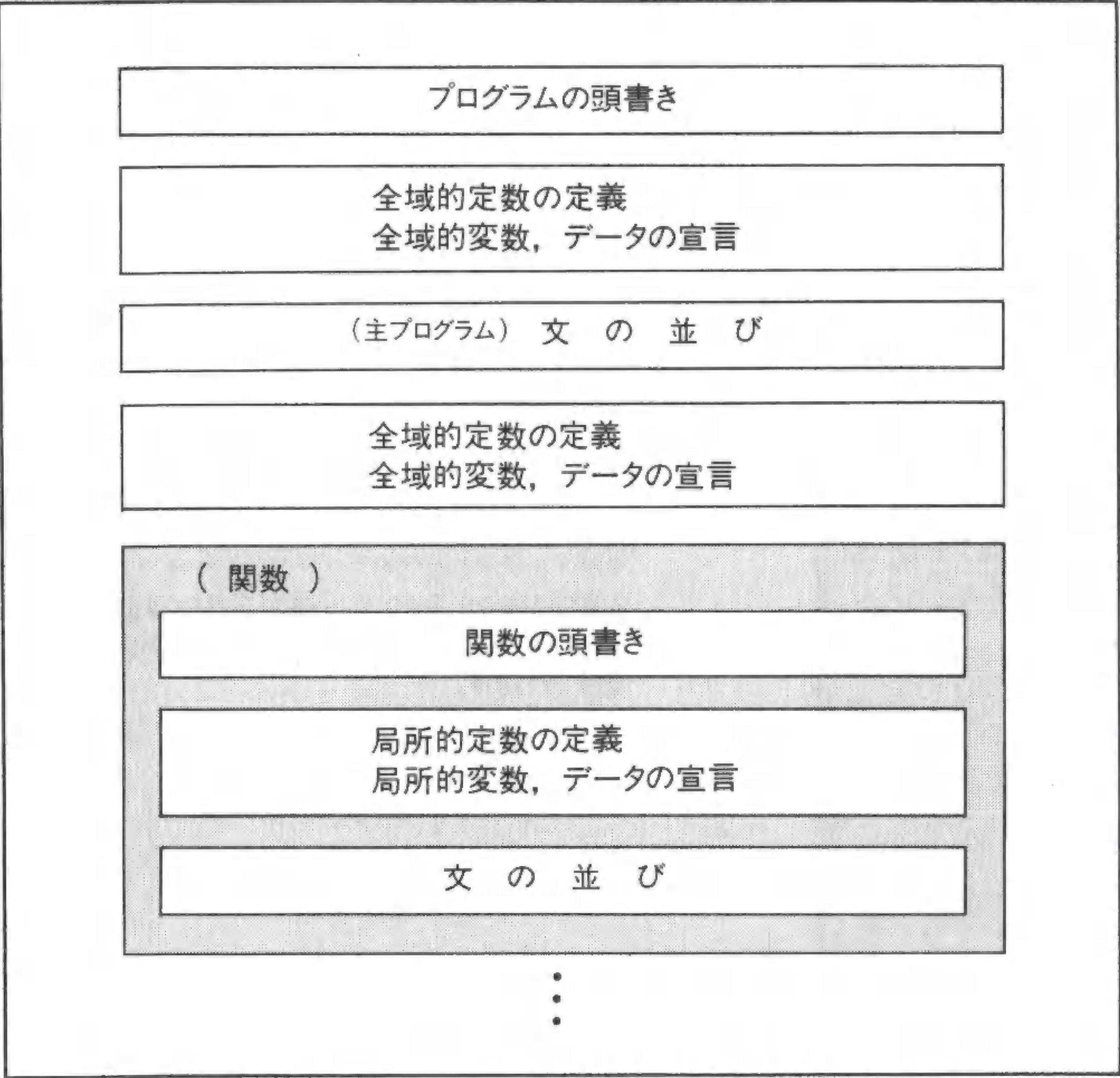
注釈はコンパイル、実行に何ら影響を与えません。

## [2] プログラムの構造

Stellar のプログラムは主プログラムといくつかの関数からなっています (図 4-2)。



図4-2 プログラムの構造



(1) プログラムの頭書き

Stellar のプログラムは必ず頭書きから始まらなければなりません。

頭書きではプログラムに名前をつけ、コンパイラに対する指示を与えます。この指示はオプション・スイッチといい、次の四つがあります（英字は小文字でも大文字でも可）。

%P：アドレス …オブジェクト・コードのロード開始アドレスを指定する。

%d：アドレス …変数やワーク・エリアの先頭アドレスを指定する。

%s：アドレス …スタックのボトム・アドレスを指定する。

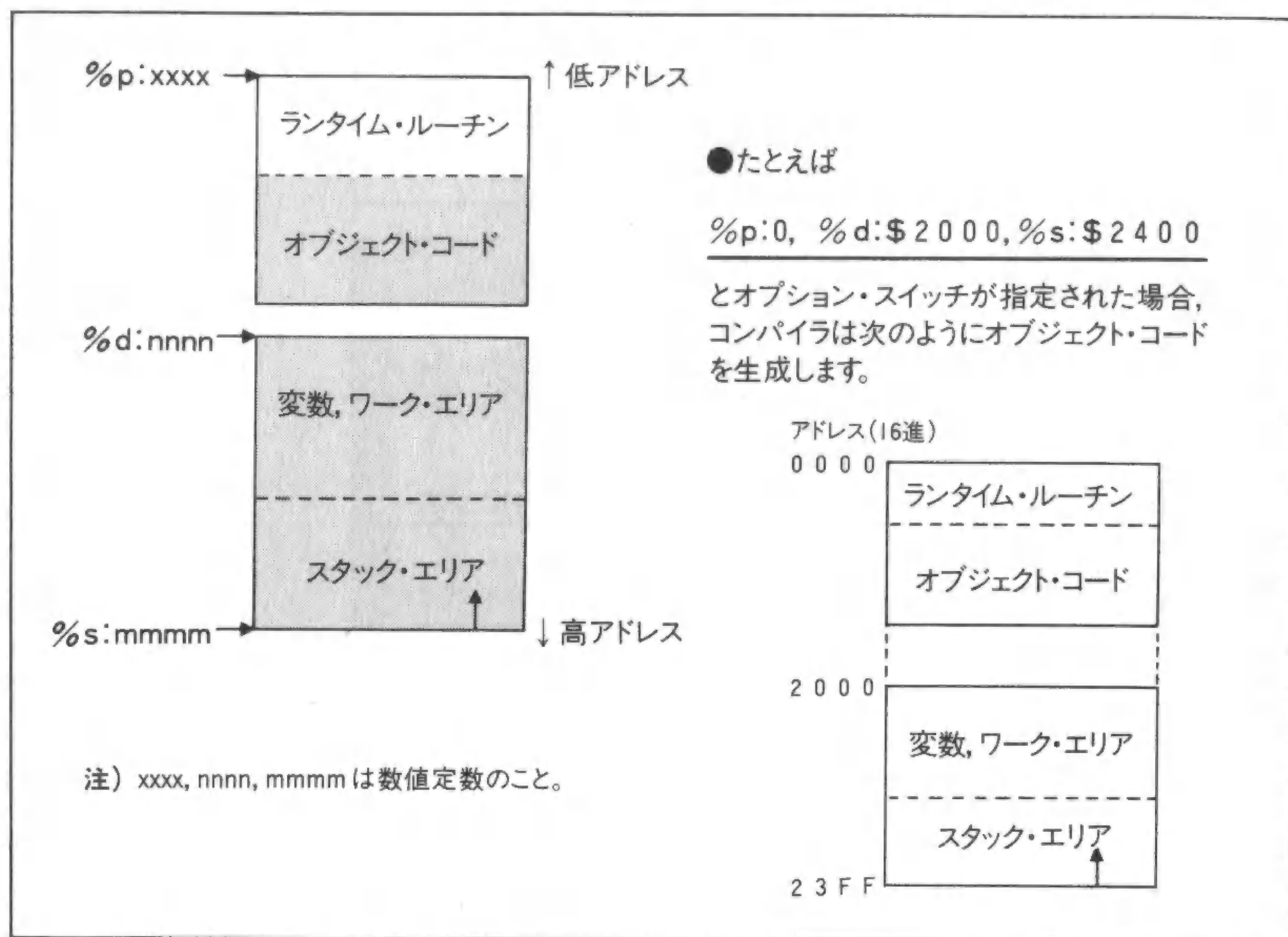
%debug …デバッグ・モードでコンパイルする。

%P、%d、%s はコンパイル時のアドレスを指定するもので、各アドレスは図 4-3 のように対応します。指定されな



いとデフォルトのアドレスになります。デフォルトは%P : \$b000, %d : \$d900, %s : \$f900となります。  
% debug を指定しないと、コンパイラはデバッグのための文を注釈として扱います。

図4-3 オプション・スイッチ%p, %d, %sとオブジェクト・プログラム



例

```
prog TEST1( ) ;
```

…プログラム名は TEST 1, オプション・スイッチの指定なし。

```
prog ut1 (% p : $ 9000, % d : $ A000, % s : $ F000) ;
```

…プログラム名は ut 1, オブジェクト・コードは 16 進数で 9000 番地から, 変数, ワーク・エリアは A000 番地, スタック・ボトムは F000 番地。

```
prog fxn (% debug, % d : 39168) ;
```

…プログラム名は fxn で, コンパイルはデバッグ・モード, オブジェクト・コードは 39168 (16 進数で 9000) 番地から出力。



(2) 定義と宣言

定義と宣言は、定数定義部、変数宣言部、データ宣言部、およびコンパイル制御文からなります。

コンパイル制御文については [3] で解説していますので、そちらを参照してください。

定数定義文はアセンブラ言語の EQU 命令に相当するもので、定義された名前のことを定数名といいます。定数名はプログラム中では数値定数、文字定数と同等に扱われます。

定数名で定義できるのは、定数式で計算された 16 ビット長の値です。定数式とは定数（数値や文字、あるいは定義済みの他の定数名など）同士を加減した式のこと、計算は常に 16 ビット長で行われオーバーフローは無視されます。

cons

cl := 120,

xyz := \$af,

c2 := cl + 'a' - xyz;

↑

↑

↑

定数名clを

定数名xyzを

定数名c2を

120と定義

\$afと定義

cl + 'a' - xyz

の計算結果の値と定義

変数宣言部は、プログラム中で使う変数を宣言する部分で、使われる変数はすべてここで宣言しておかなければなりません。ここでは変数にアドレスを割りつけるだけで、初期値は出力しません。ですから、初期値が必要な変数はプログラム中で値を設定しなければなりません。図 4-4 は変数の宣言とメモリへの割りつけを示した例です。

データ宣言部は定数をメモリ上に記憶するためのもので、記憶した定数にはデータ名がつけられます。データ名は代入できないことを除き、プログラム中で変数と同等に使うことができます。

定数の前の byte や word は、記憶がバイト単位なのかワード単位なのかを指定するものです。バイトで記憶する場合は byte の指定は省略できます。また、word の代わりに # を使うこともできます。



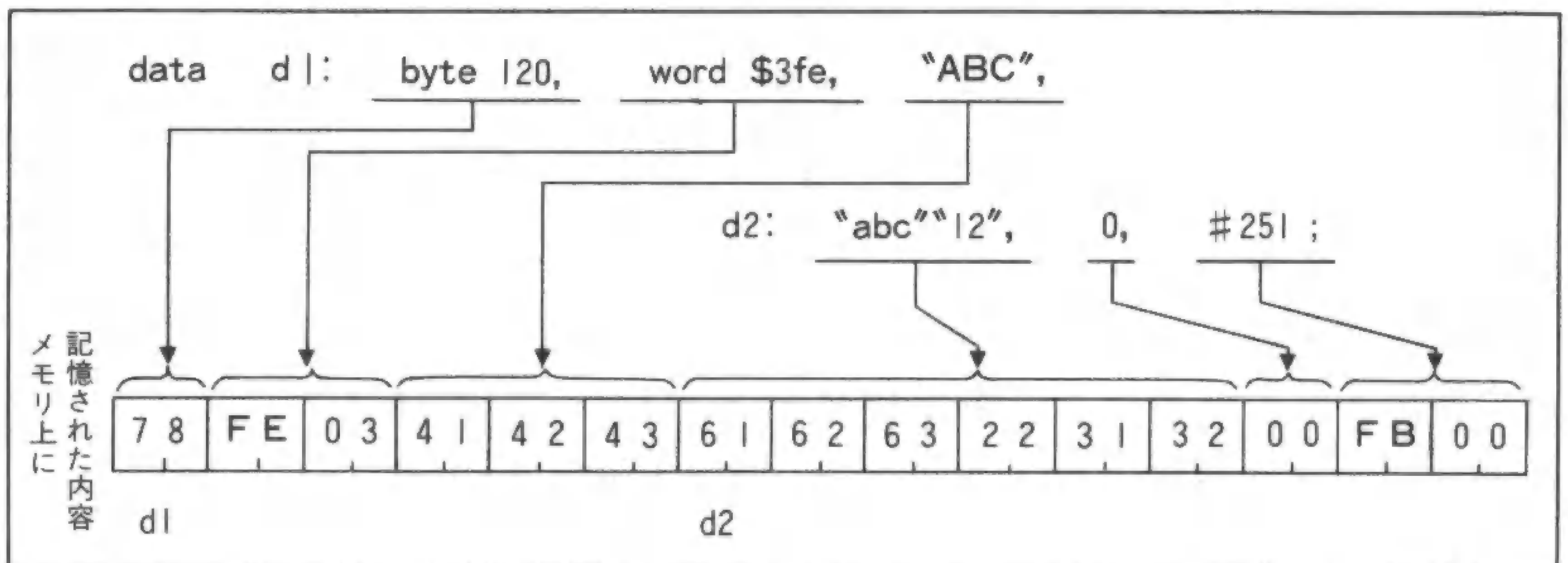
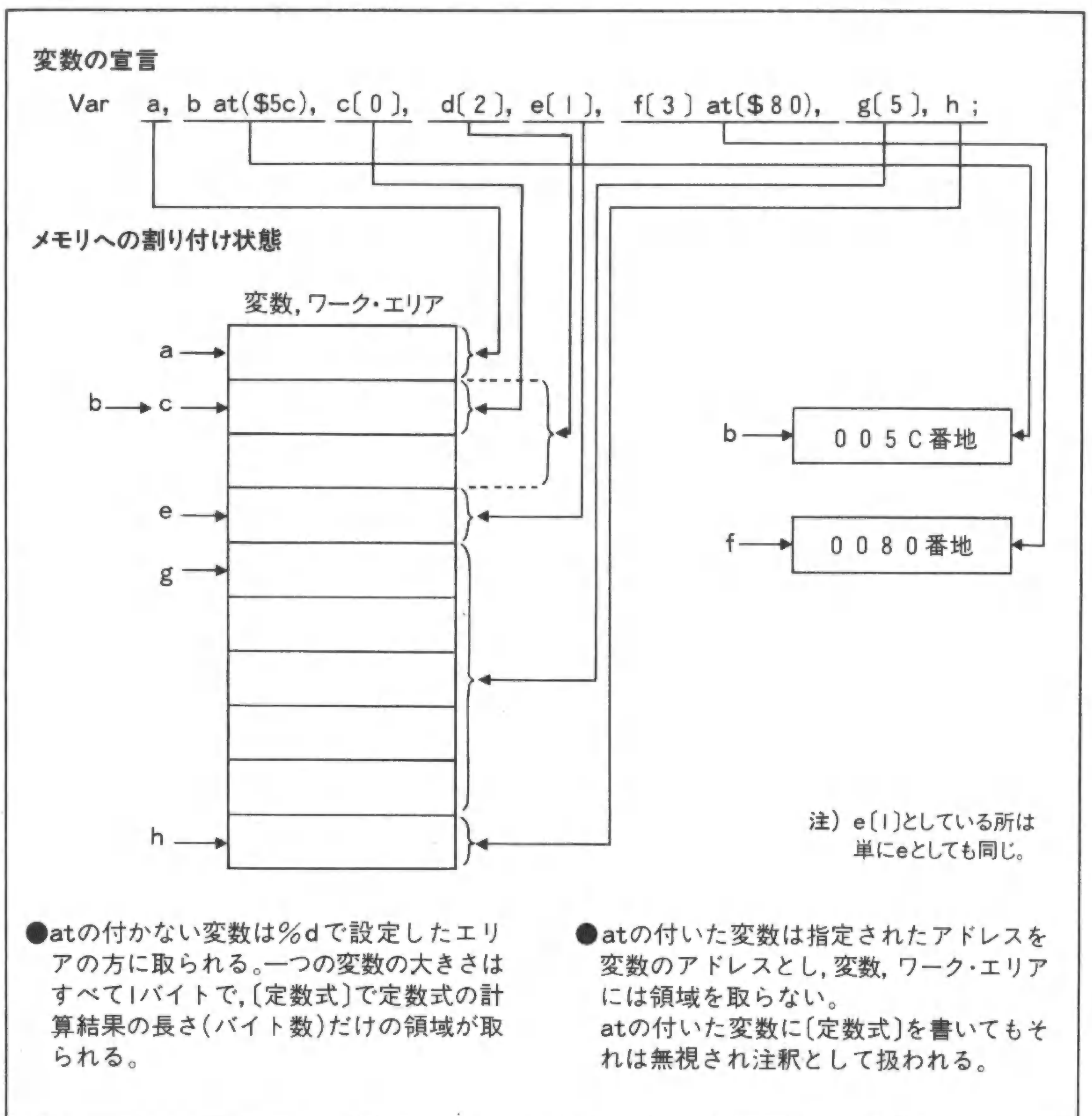


図4-4 変数の宣言とメモリへの割り付け例





(3) 主プログラム

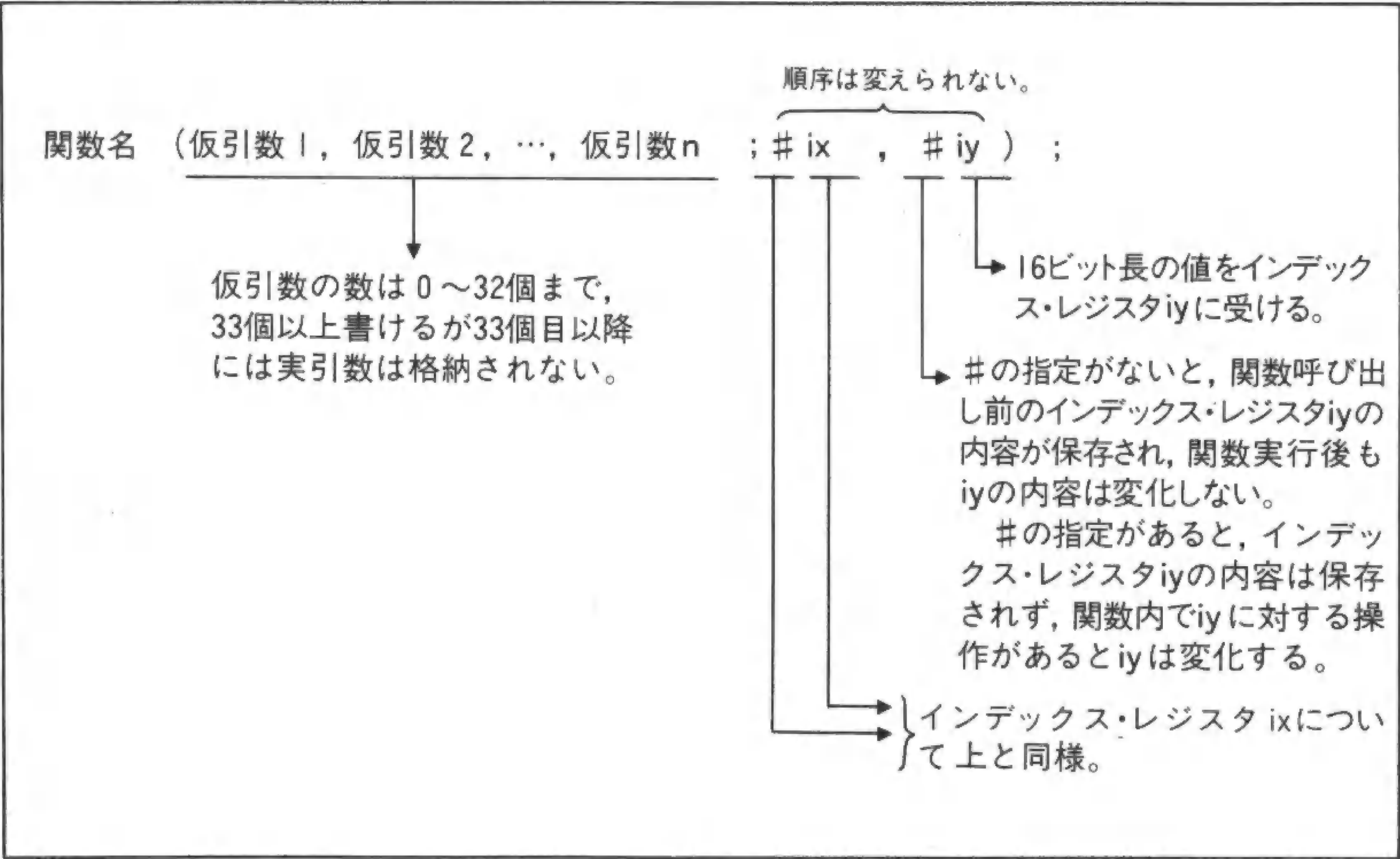
Stellar のプログラムは必ず主プログラムから実行されます。主プログラムは一つの複合文として書かれます。

(4) 関数の頭書き

関数の頭書きは、ここから関数が始まることを宣言するためのもので、一つの関数は関数の頭書きとブロックにより構成されています。

再帰的に呼び出す関数は、関数名のまえに recursive と書かねばなりません。recursive と書かれている関数は、呼び出されたとき局所的な変数の値をスタックに保存します。

Stellar では、関数に実引数の値を渡します。引数には 8 ビット長と 16 ビット長の二種類があり、8 ビット長の値は仮引数（局所的な変数）に、16 ビット長の値は ix あるいは iy のインデックス・レジスタに格納されます。関数の頭書きに書かれる仮引数やインデックス・レジスタ名は次のような意味を持っています。





## (5) ブロック

ブロックは、定義と宣言、および一つの複合文により構成されます。

ブロック内で定義あるいは宣言された定数名、変数名、データ名は局所的な名前として扱われ、有効範囲はその関数内だけです。

## (6) 名前の有効範囲

Stellar では名前の有効範囲を次のように決めています。

### ①局所的な名前

仮引数、およびブロック内で定義された定数名、変数名、データ名は局所的な名前です。同一関数名でのみ有効です。主プログラムや他の関数からは使えません。全域的な名前と同じ名前を局所的な名前として定義、宣言した場合、その関数内では局所的な名前を優先し、全域的な名前のほうは参照できません。

### ②全域的な名前

局所的な名前以外はすべて全域的な名前です。全域的な名前は、主プログラムや関数の別なくプログラム全体で共通に使えます。全域的な名前には関数名や定数名、変数名、データ名があり、関数名以外は定義、宣言された以後有効になります。関数名については、書かれている場所によらず、すべての場所でも有効です。

## [3] 文 (ステートメント)

文は実際に計算や仕事をするもので、図 4-1 に示したように 15 種類あります。式は次の [4] で解説するとして、ここでは式以外の文について解説します。

文にはラベルをつけることができます。ラベルは主プログラムや各々の関数内でのみ有効な局所的な名前です。

### (1) 空文

何の動作もしない文のことです。



## (2) 複合文

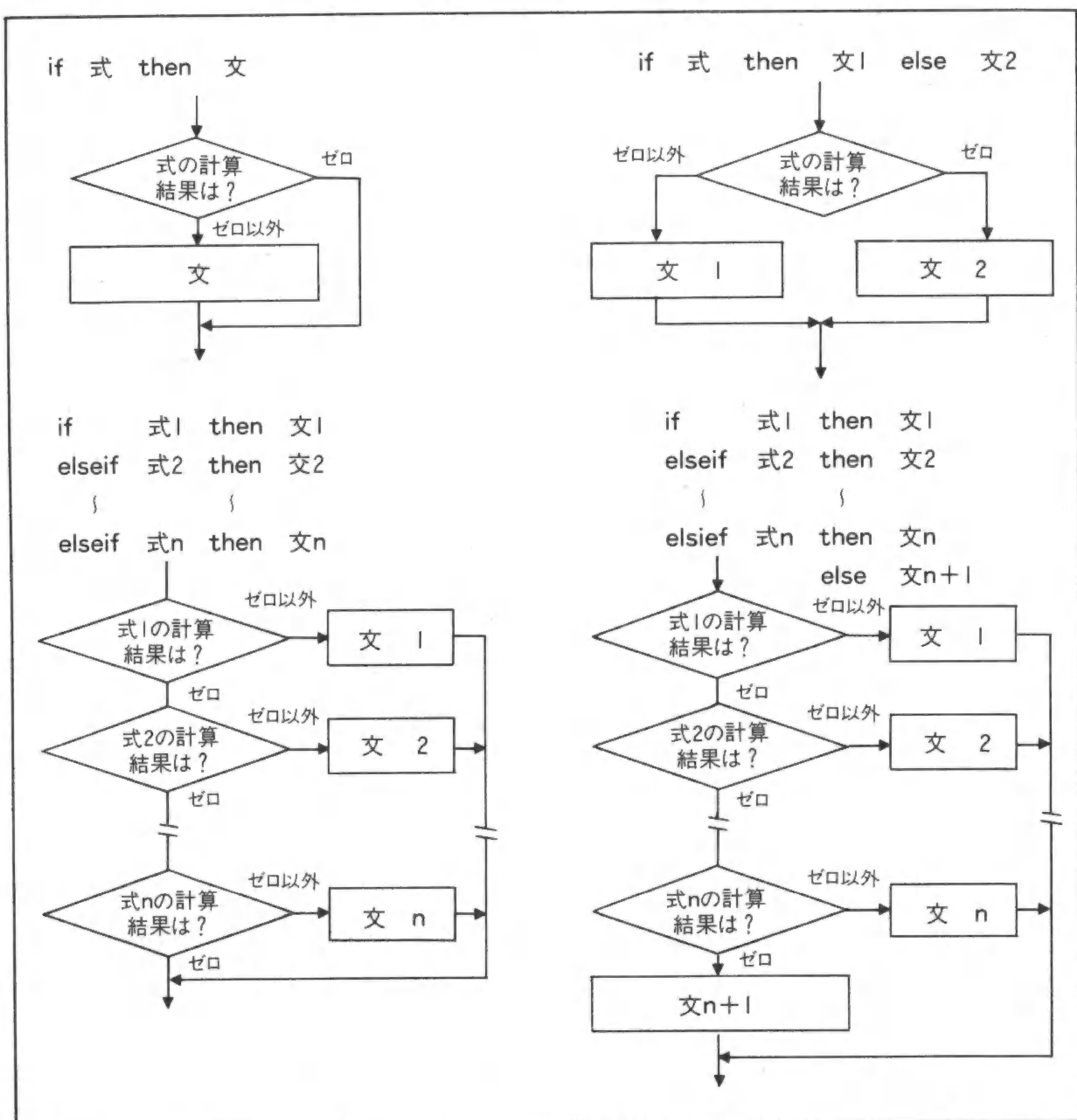
複合文はいくつかの文をまとめて一つの文として扱うもので、文が書かれている順に実行が行なわれます。

## (3) if 文

if 文は式の計算結果がゼロ以外のときを真、ゼロのときを偽としています。真のときは then の次の文を、偽のときは else の次の文を実行します。else の次の文が if 文のときは、elseif とすることが出来ます。

図4-5はif文のいくつかの形式を流れ図で表わしたものです。

図4-5 if文



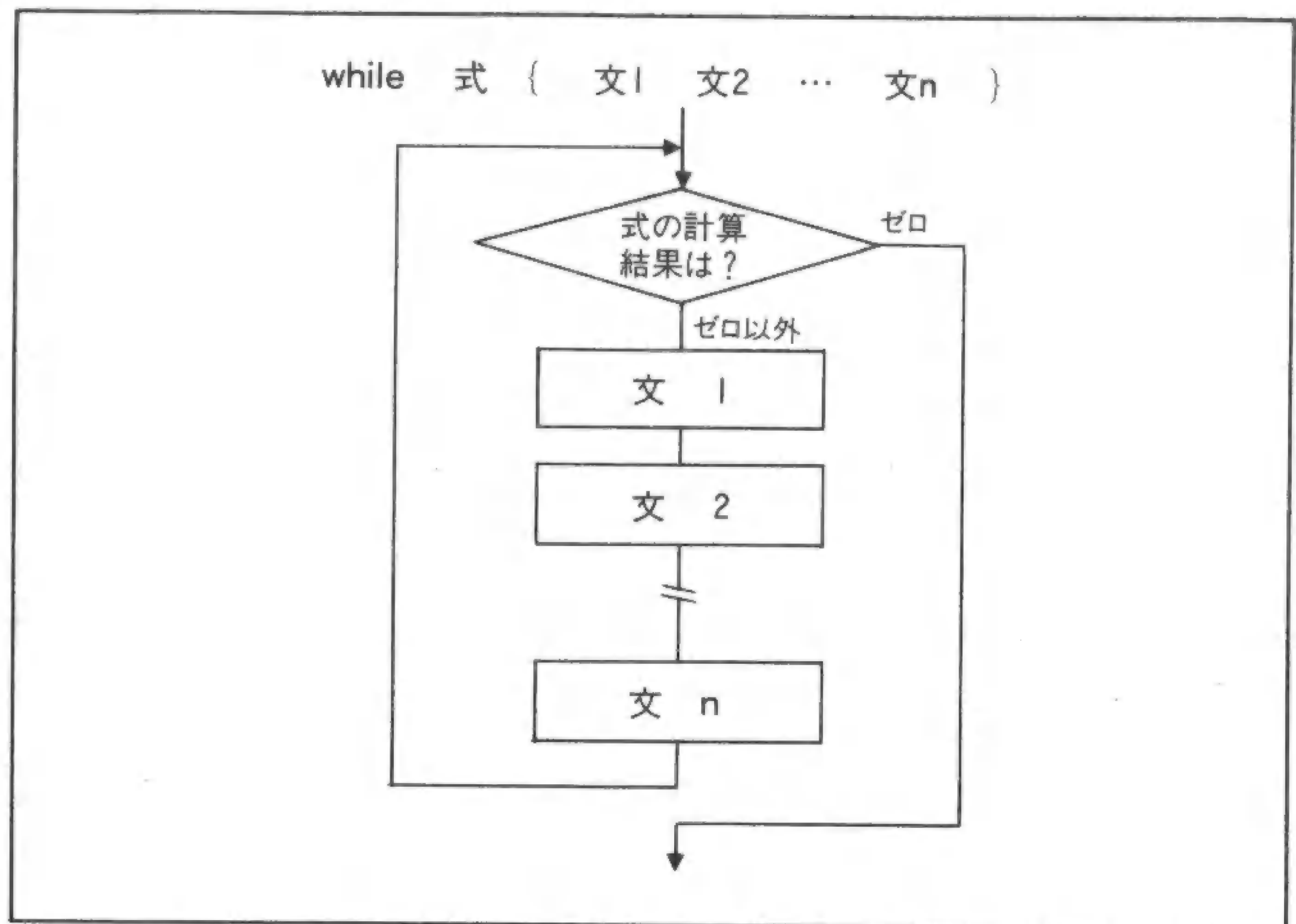


#### (4) while 文

繰り返しのための文の一つです。

while 文は、式の計算結果がゼロになる（偽になる）まで式の後ろの複合文を繰り返し実行します。while 文では始めに式の計算結果を調べるため、場合によっては後ろの複合文が一度も実行されないことがあります（図4-6）。

図4-6 while文



#### (5) until 文

until 文も繰り返しを行なうための文です。

until 文はまず until の前の複合文を実行し、その後に式の計算を行ないます。そして結果がゼロ（偽）ならさらにもう一度 until の前の複合文を実行し、結果がゼロ以外（真）なら繰り返しを終えます（図4-7）。

#### (6) for 文

for 文は流れ図では図4-8のようになります。for の次の変数を特に制御変数といいます。:= の後の式を初期値、to の後の式を終値として、制御変数が

制御変数の値 > 終値



図4-7 until文

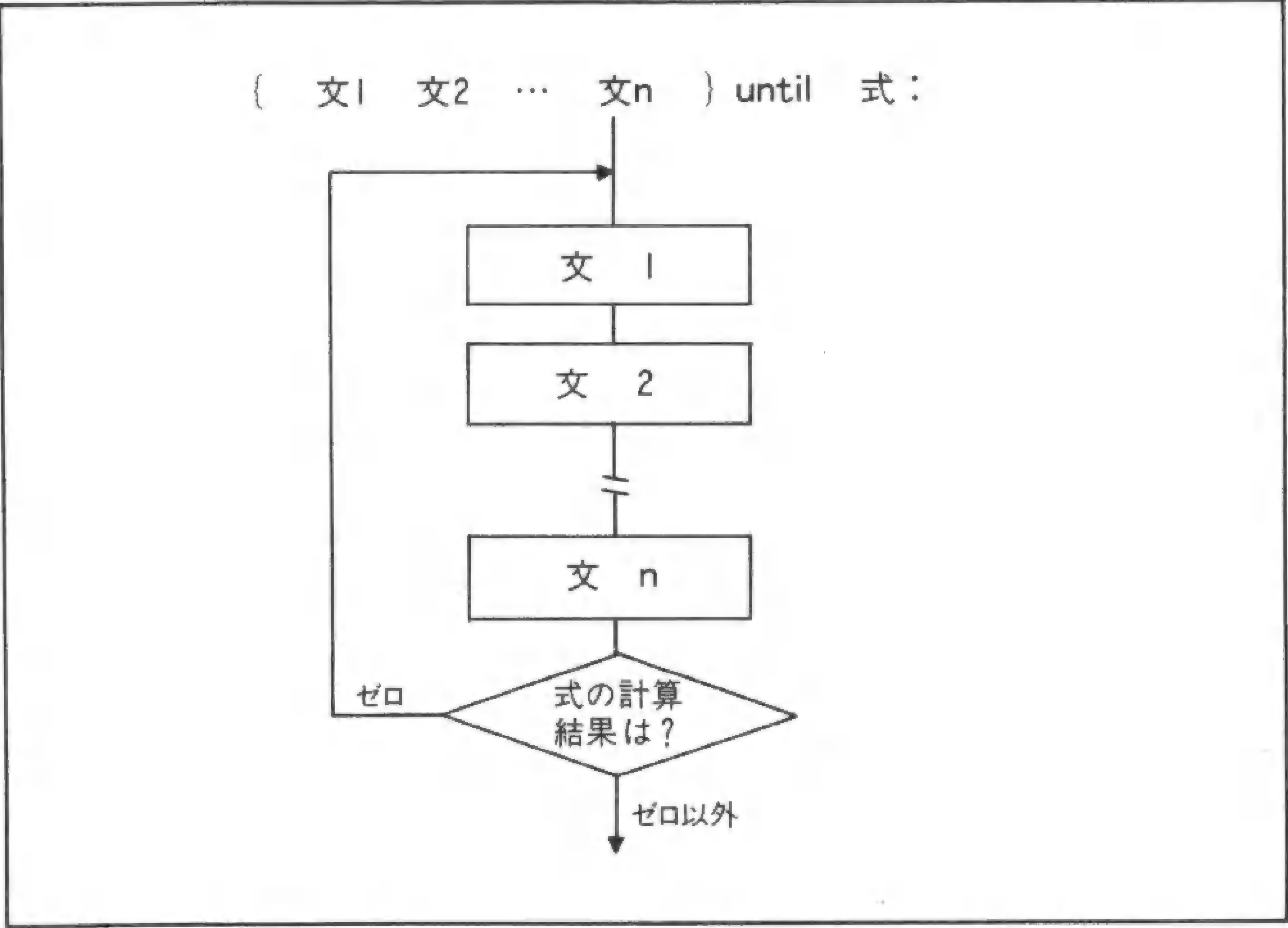
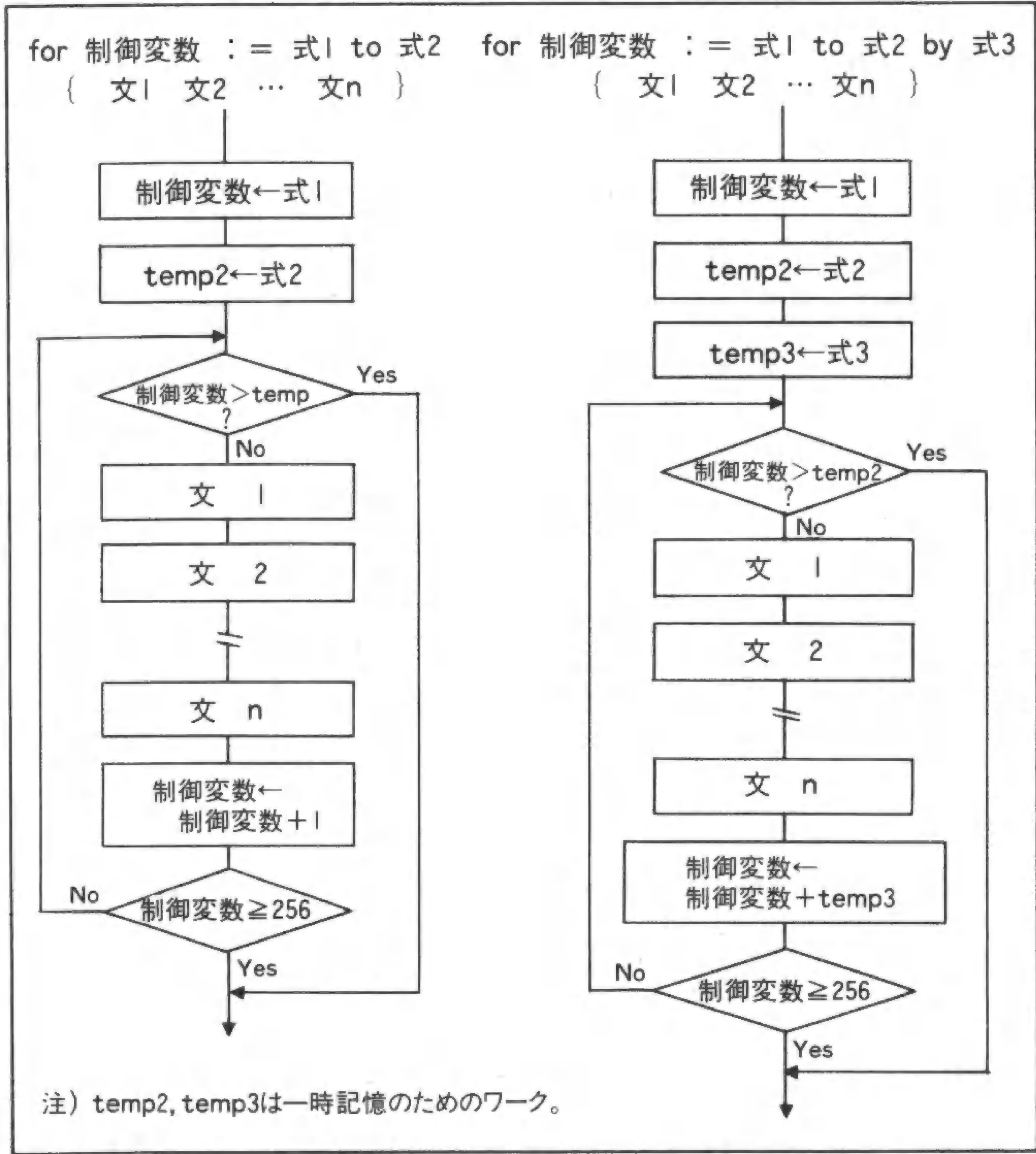


図4-8 for文





あるいは

制御変数の値 $\geq$ 256
-------------------

となるまで繰り返します。このとき、by 式の増分が指定されていれば制御変数にその増分を加算し、増分の指定がなければ制御変数を+1します。for 文では初期値、終値、増分を示す各式を繰り返しの前に計算します。初期値は制御変数に代入され、終値、増分はメモリ上の一時記憶域に記憶します。繰り返しの途中はその領域に記憶した値を使います。Stellar の場合、変数は 1 バイト長だけなので、繰り返し回数の最大は

for 変数名 : = 0 to 255
----------------------

としたときの 256 回です。

#### (7) loop 文

loop 文は指定された回数を単純に繰り返すだけの文です。loop 文を流れ図で表わすと図 4-9 のようになります。単純な繰り返しなら for 文よりも loop 文のほうがコンパイル後のオブジェクト・コードが少なくてすみます。特に制御変数名の代わりに#を使うと b レジスタでループするようなオブジェクト・コードがつくられるため、さらに効率がよくなります。

#### (8) goto 文

ラベルが書かれている文に実行を移します。

#### (9) exit 文

複合文や while 文、until 文、for 文、loop 文の繰り返しから抜け出るときに使います (図 4-10)。

#### (10) return 文

関数の実行を終わり、呼び出した主プログラムや関数に戻



るときに使います。関数の最後の return 文は省略しても自動的に挿入されます。また、主プログラム内の return 文は stop 文と同じ意味です。

図4-9 loop文

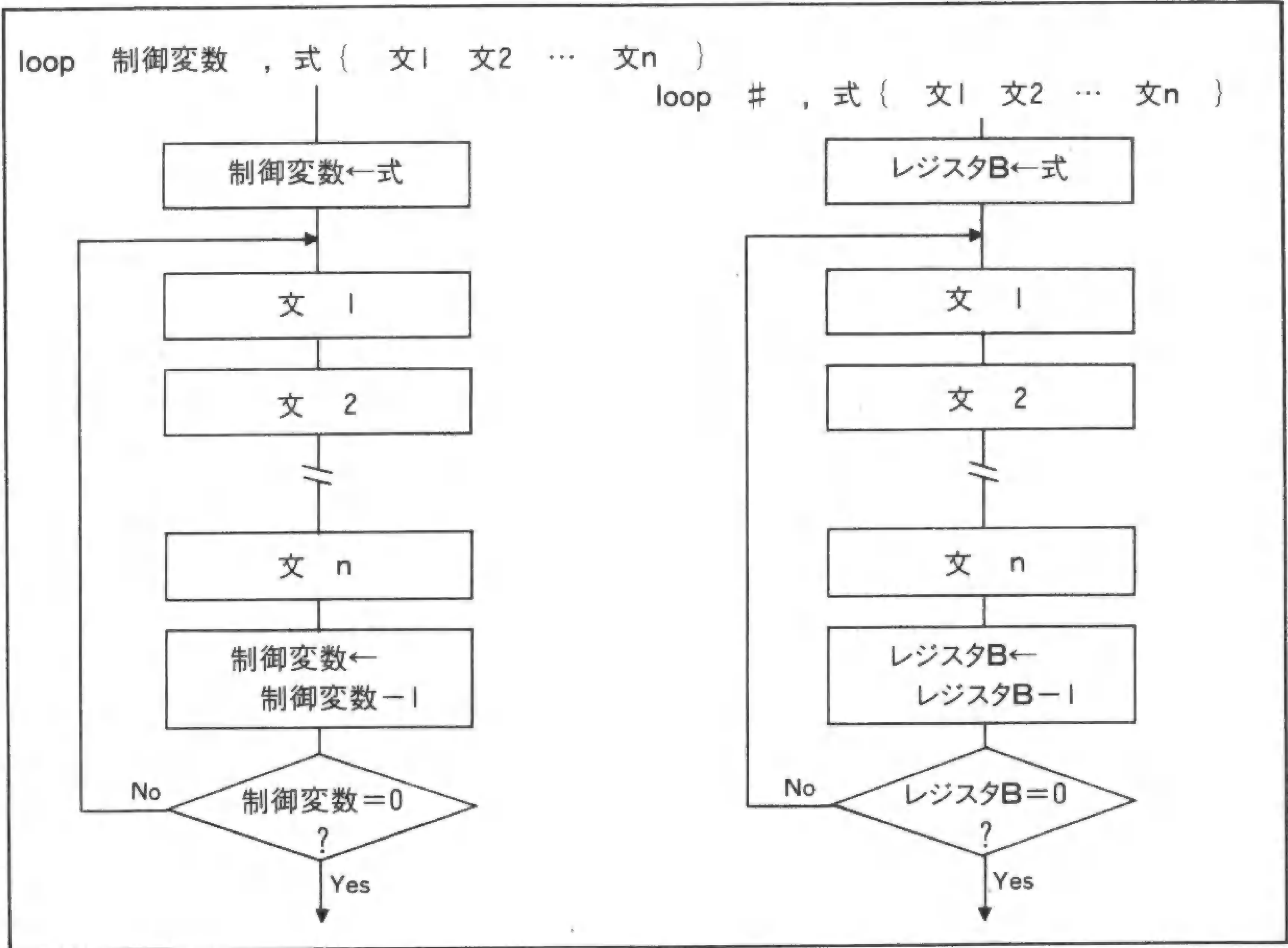
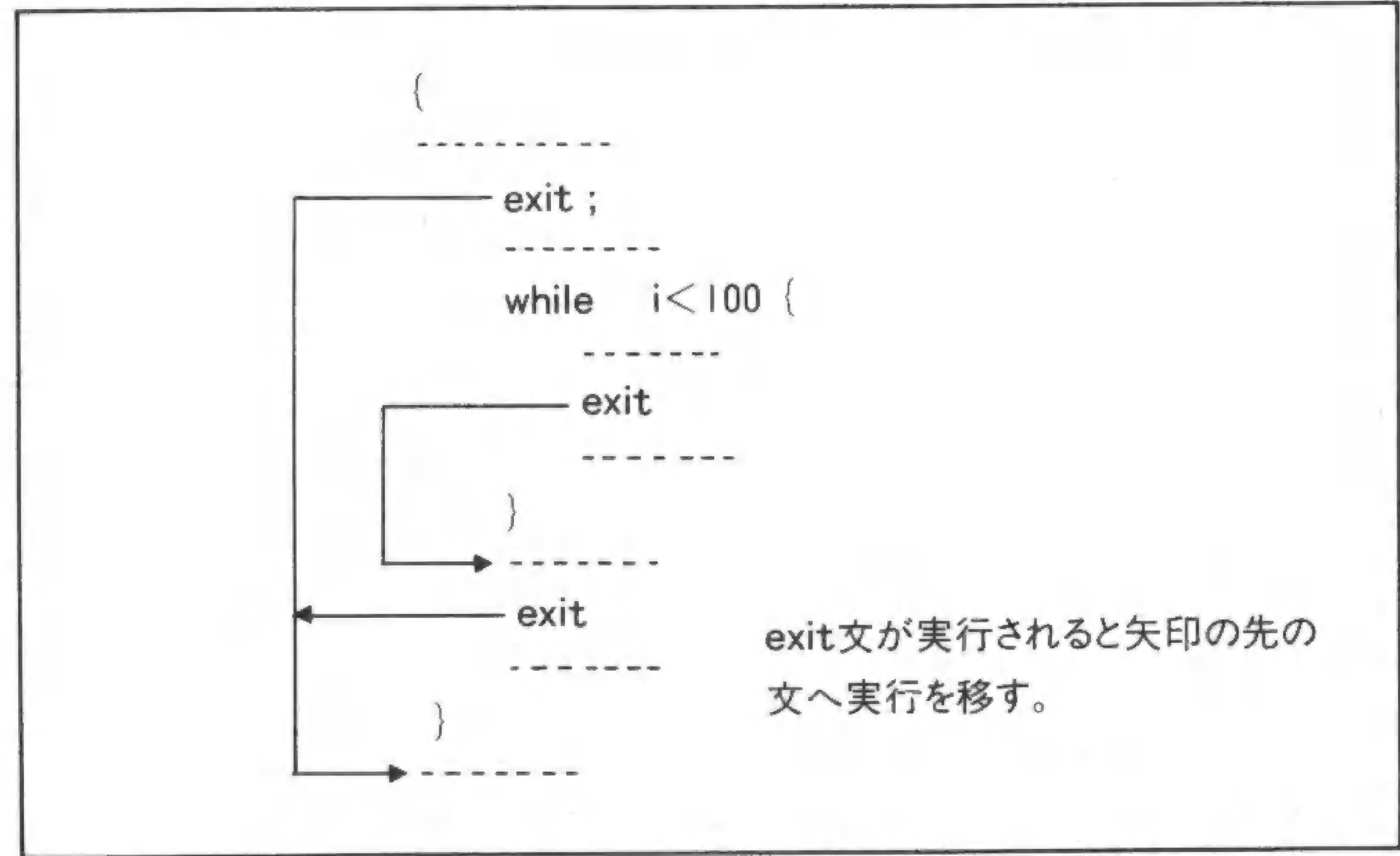


図4-10 exit文の動作例





### (11) stop 文

プログラムを終了させるための文です。主プログラムの最後の stop 文は省略しても自動的に挿入されます。

### (12) inline 文

inline 文はプログラム中に機械語命令やデータを入れるための文です。

### (13) インデックス操作文

インデックス操作文は二つのインデックス・レジスタ  $ix$ ,  $iy$  を操作するための文で、次に示すような 5 個の文があります。

#### ① set 文

set 文はインデックス・レジスタに定数や値をセットする文です。

定数とインデックス・レジスタは 16 ビット長で扱われます。式は 8 ビット長で計算され、インデックス・レジスタの値 (16 ビット長) に加算または減算されます。

例

```
set ix := $3F2C ;  
set iy := 258 + a / 3 ;  
set ix := iy - b * 2 ;  
set ix := ix + 3 ;  
set iy := ix ;
```

#### ② ldx 文

ldx 文はインデックス・レジスタに変数の値をロードする文です。

この文によりインデックス・レジスタの下位 8 ビットには変数の値が、上位 8 ビットには変数の次のアドレスの値がロードされます。ldx 文と次の stx 文だけが変数を 16 ビット長で扱います (図 4-11)。

#### ③ stx 文

stx 文はインデックス・レジスタの値を変数にストアする



文です。

この文によりインデックス・レジスタの下位8ビットが変数へ、上位8ビットが変数の次のアドレスへストアされます (図4-12)。

#### ④ inx 文, dex 文

inx 文は指定したインデックス・レジスタの値を+1し、dex 文は-1します。

図4-11 ldxの例

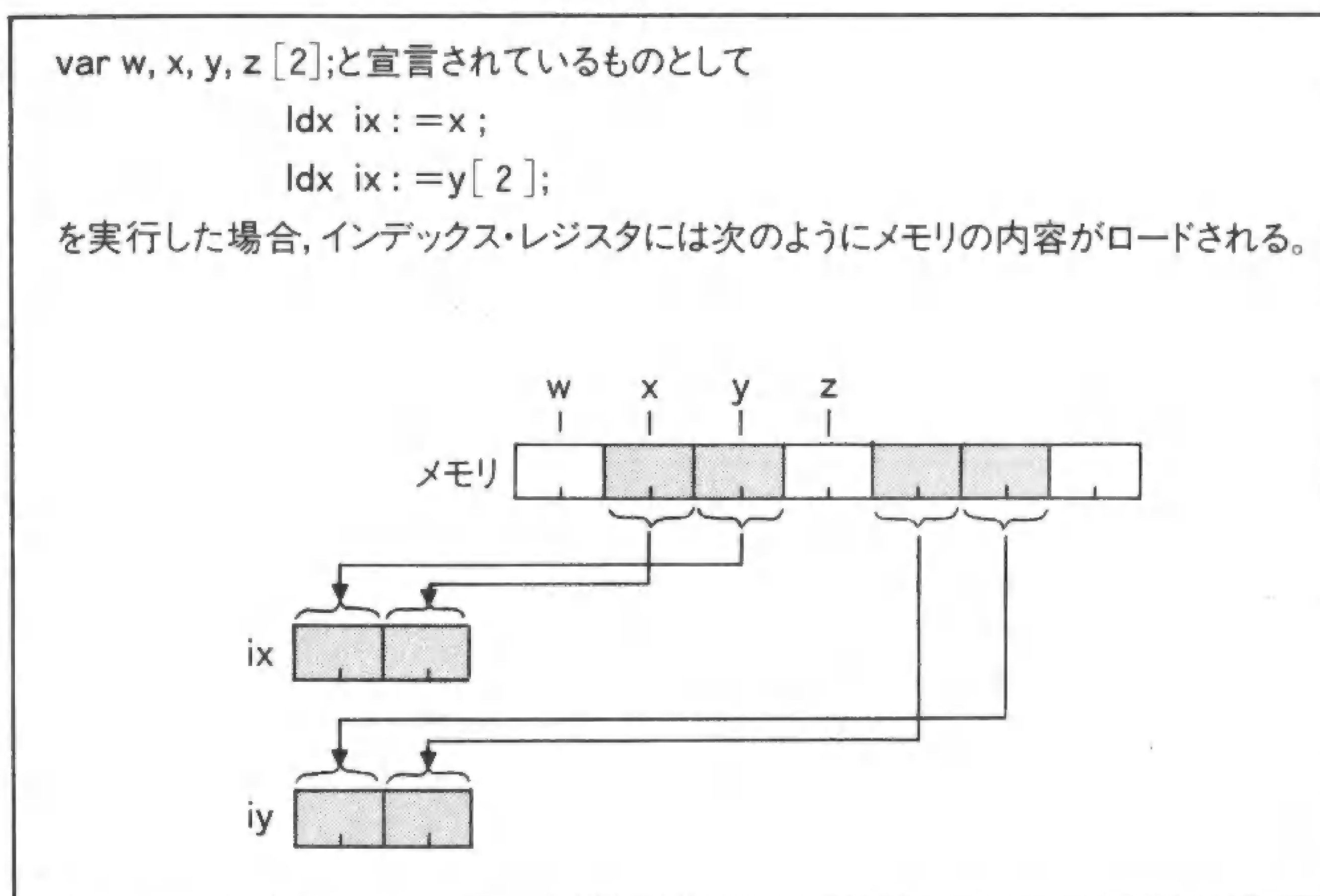
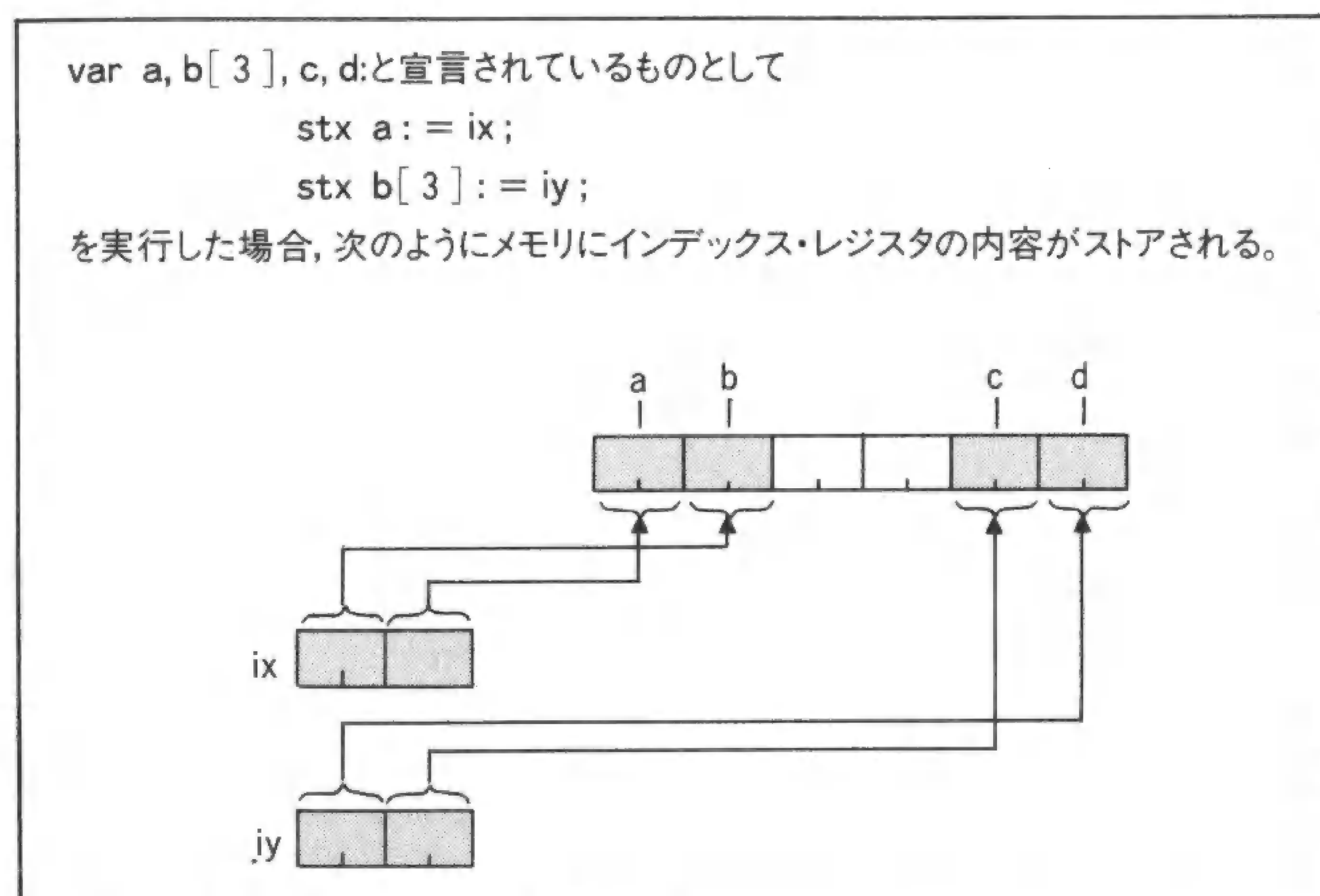


図4-12 stxの例





#### (14) コンパイル制御文

コンパイル制御文には、% tron, % troff, % break, % include の四つの文があります。

% tron, % troff, % break はプログラムのデバッグのための文で、デバッグ・モードでコンパイルしたときだけ有効になります。このうち、% tron, % troff はプログラムのトレースに関する文で、% tron を指定するとプログラムの実行状態が追跡できるようなオブジェクト・コードを生成し、% troff が指定されるまで続きます。% tron によって生成されるコードは、実行した行番号あるいは関数名を表示するもので、行番号は

[ 行 番 号 ]
-----------

関数名は

{ 関 数 名 }
-----------

と表示します。

% break はプログラムを一時的に中断する文で、これを実行すると

** break in 行番号
-----------------

と表示して実行を中断します。中断したプログラムは再開させることができます。

% include は指定されたファイルをソース・プログラムの一部として読み込むための文で、ライブラリなどを読み込むために使います。

#### [ 4 ] 式

Stellar の式はすべて 8 ビット長で演算されます。演算は論理、関係、算術の三つに分類されます。

##### (1) 演算子と演算

演算子は 25 個あり、それぞれ次に示すような演算を行います。



①論理演算子

論理演算子はブール演算やビット操作を行ないます。

or !	…論理和
xor	…排他的論理和
and &	…論理積
not ~^	…論理否定

論理演算は各ビットごとに演算が行なわれます。

②関係演算子

関係演算子は二つの値を比較するためのものです。

=	…等しい
<>	…等しくない
>	…大きい
>=	…大きいか等しい
<	…小さい
<=	…小さいか等しい

関係演算子は符号なしの2進整数で比較を行ない、結果が真のとき255、偽のとき0の値が得られます。

③算術演算子

算術演算子には次のものがあります。

+	…加算
-	…減算
単項+	…2の補数とはらない（何もしない）
単項-	…2の補数をとる
*	…乗算
/	…除算
%	…剰算
plus	…キャリーフラグを含めた加算
minus	…キャリーフラグを含めた減算



算術演算は符号なしの2進整数で行なわれます。

#### ④ その他の演算子

演算子としてはその他に、シフトと代入があります。

<<…左シフト (<<の右辺の値で左辺の値を論理左シフトする)

>>…右シフト (>>の右辺の値で左辺の値を論理右シフトする)

:=…代入 (:=の右辺の値を左辺へ代入する)

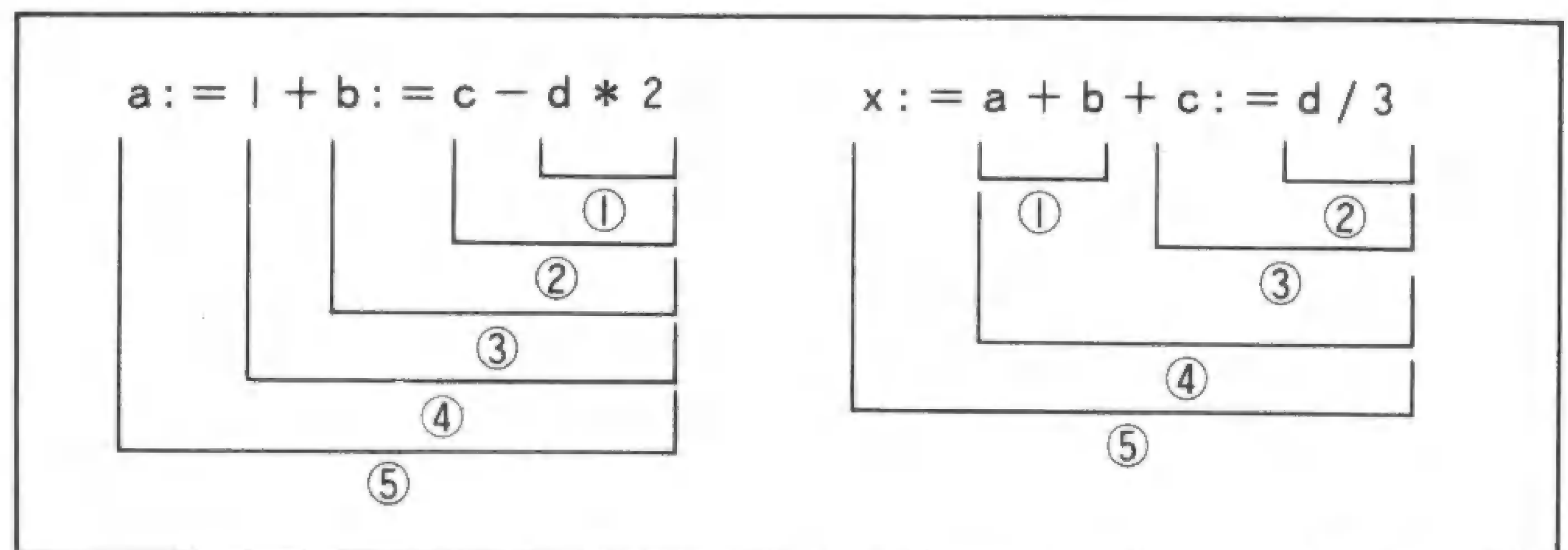
#### (2) 演算の優先順位

演算は次の優先順位によって行なわれます。

- |   |                              |
|---|------------------------------|
| 高 | 1. ( ) や [ ] の中の式, :=の右辺, 関数 |
|   | 2. 単項の+ -                    |
|   | 3. * / % << >>               |
|   | 4. + - plus minus            |
|   | 5. = <> > >= < <=            |
|   | 6. not                       |
|   | 7. and                       |
| 低 | 8. or, xor                   |

同順位の演算は左から右へ行なわれます (図4-13)。この図で演算は①→⑤の順に行なわれます。

図4-13 演算の順位



(3) カンマ式

カンマ ( , ) で区切られたカッコの内の式は左から右へ計算され、最右の式の値がカンマ式の値になります。

例

$x := (a := 1, b := 5, a + b)$

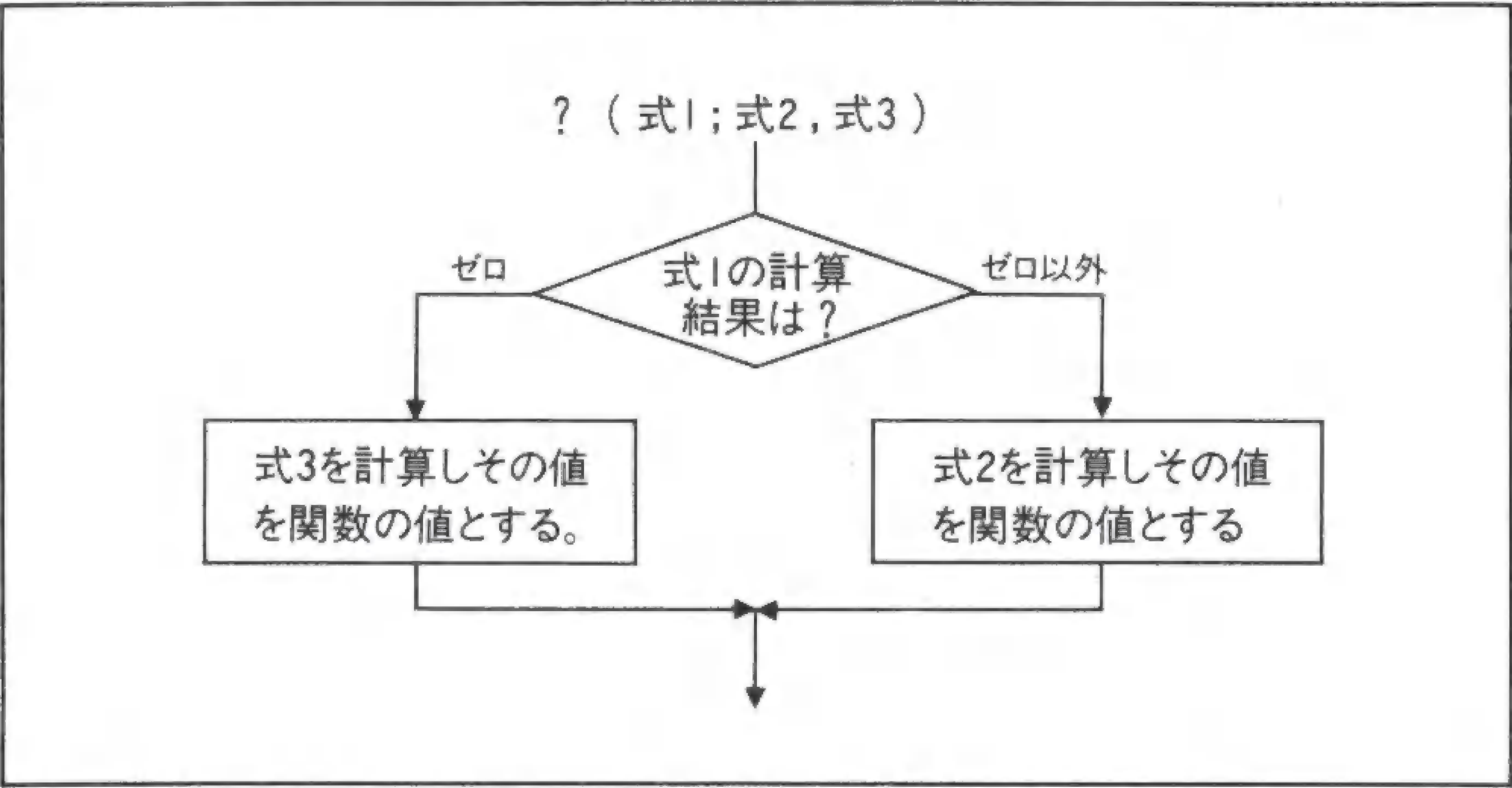
a に 1, b に 5 を代入し、 $a + b$  を計算する。 $a + b$  がカンマ式の結果となり x に代入される。つまり、x には 6 が代入される。

(4) 条件演算関数

条件演算関数とは式の中に書ける if 文というようなものです。

カッコ内には三つの式が書かれ、セミコロン ( ; ) とカンマ ( , ) によって区切られます。第一の式が条件を表わす式で、計算結果がゼロ以外 (真) のとき第二の式を計算し、関数の値とします。第一の式の計算結果がゼロ (偽) なら第三の式を計算し関数値とします。図 4-14 はこれを流れ図で表わしたものです。

図4-14 条件演算関数





例

hex 1 := ? (h := h & \$ f) < 10 ; h, h + 7)  
+ ' 0'

変数 h の下位 4 ビットを 1 文字の 16 進数に変換し、  
hex 1 へ代入する。

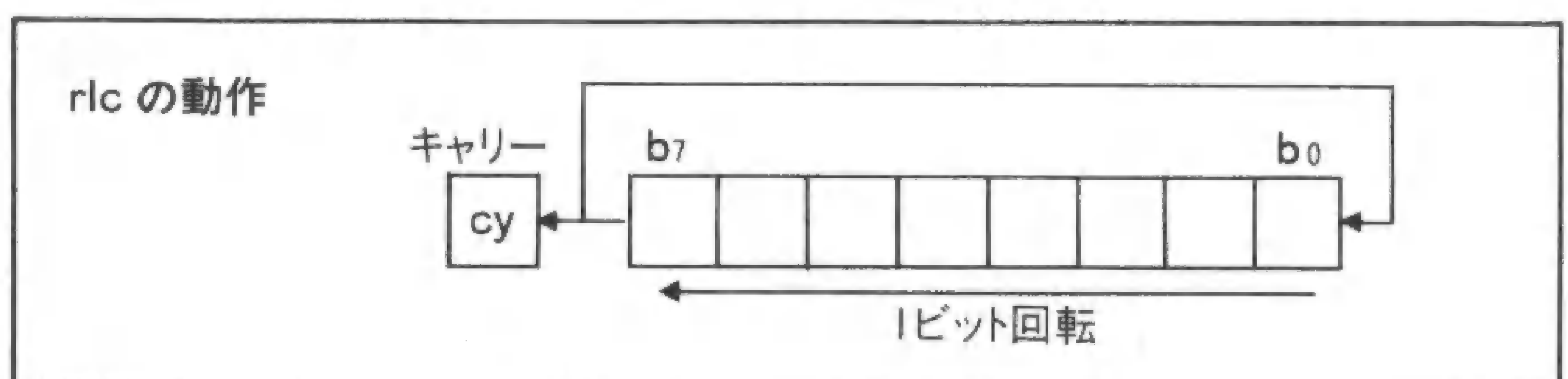
#### (5) 組み込み関数

Stellar ではつぎの 13 個の組み込み関数を持っています。

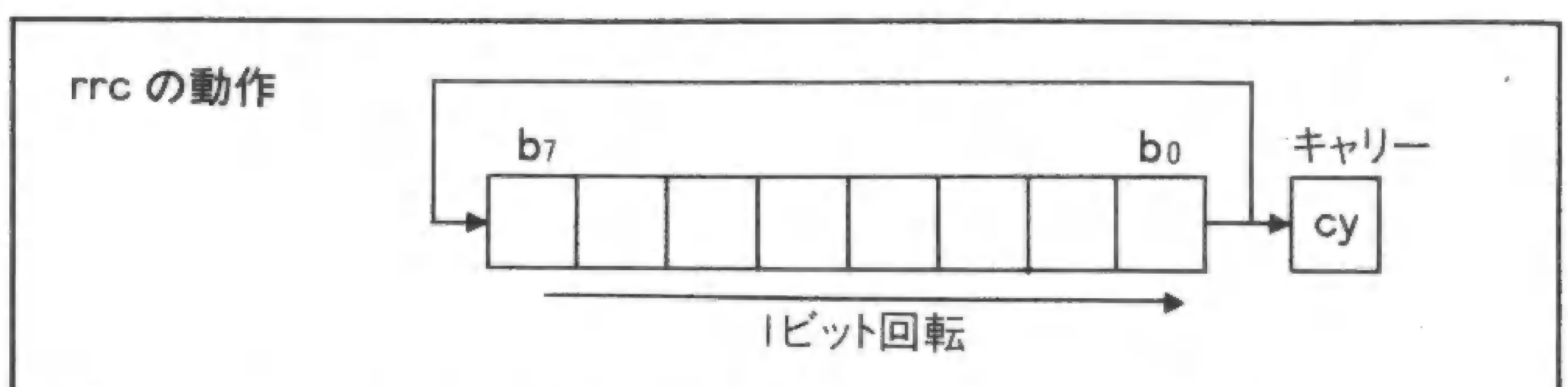
inc (変数名) ..... 変数の内容を + 1 し、+ 1 した  
値を関数の値とする。これは  
(変数名 := 変数名 + 1) と同  
じ。

dec (変数名) ..... 変数の内容を - 1 し、- 1 した  
値を関数の値とする。これは  
(変数名 := 変数名 - 1) と同  
じ。

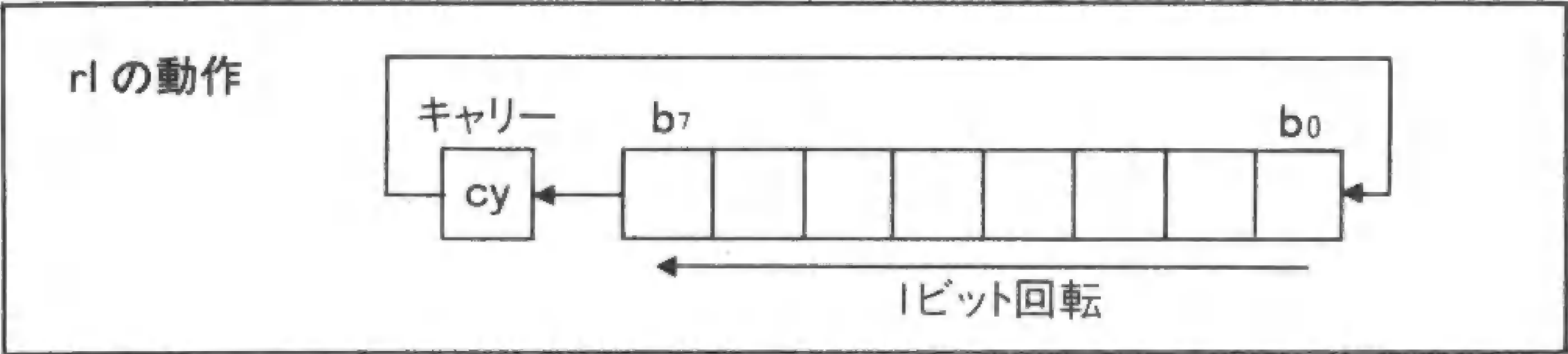
rlc (式) ..... 式の計算結果を左へ 1 ビット回  
転させる。



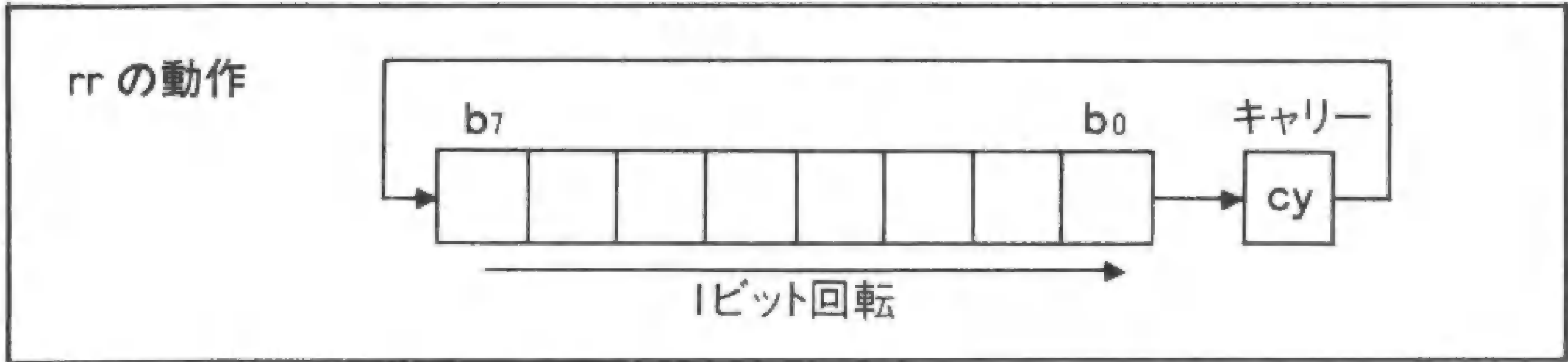
rrc (式) ..... 式の計算結果を右へ 1 ビット回  
転させる。



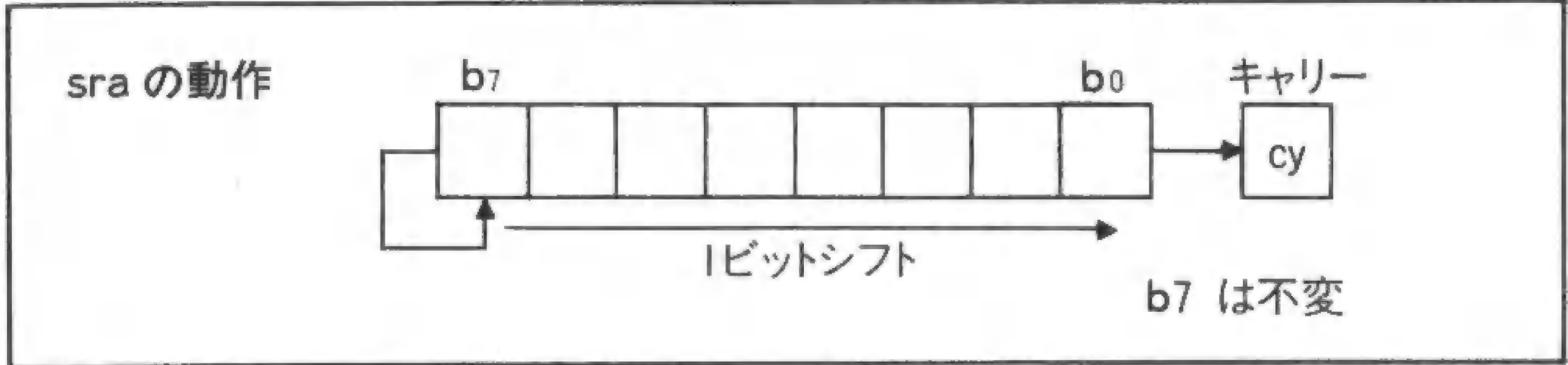
rl (式) .....式の計算結果を左へキャリーフラグも含めて 1 ビット回転させる。



rr (式) .....式の計算結果を右へキャリーフラグも含めて 1 ビット回転させる。



sra (式) .....式の計算結果を右へ 1 ビット算術右シフトする。



decj (式) .....式の計算結果を 10 進数 2 桁 (BCD) に補正する。

carry ( ) .....現在のキャリーフラグ (CY) の値を求める。CY=1 なら 255, CY=0 なら 0 が関数値。

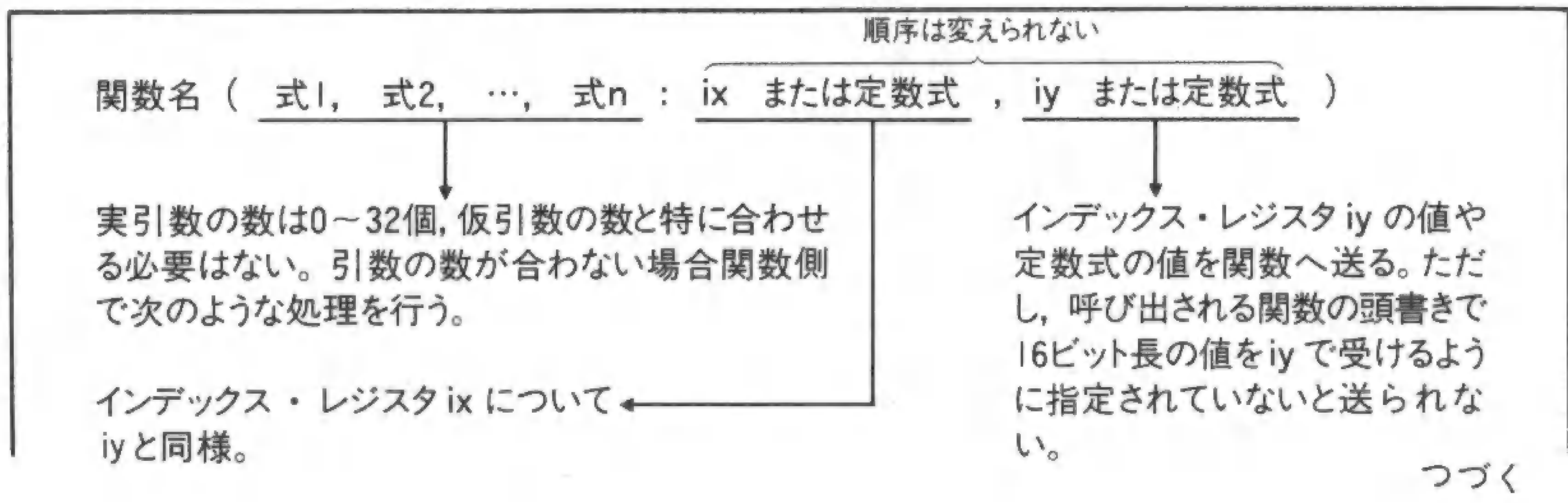
carry (式) .....式計算後のキャリーフラグの値を求める。CY=1 なら 255, CY=0 なら 0 が関数値。



- zoro (式) .....式計算後のゼロ・フラグの値を  
求める。Z = 1 なら 255, Z =  
0 なら 0 が関数値。
- sign ( ) .....現在のサイン・フラグ (S) の  
値を求める。S = 1 なら 2 5  
5, S = 0 なら 0 が関数値。
- sign (式) .....式計算後のサイン・フラグの値  
を求める。S = 1 なら 255,  
S = 0 なら 0 が関数値。
- parity ( ) .....現在の P / V フラグの値を求め  
る。P / V = 1 なら 2 5 5,  
P / V = 0 なら 0 が関数値。
- parity (式) .....式計算後の P / V フラグの値を  
求める。P / V = 1 なら 255,  
P / V = 0 なら 0 が関数値。
- overflow ( ) ..... parity ( ) と同じ。
- overflow (式) ..... parity (式) と同じ。

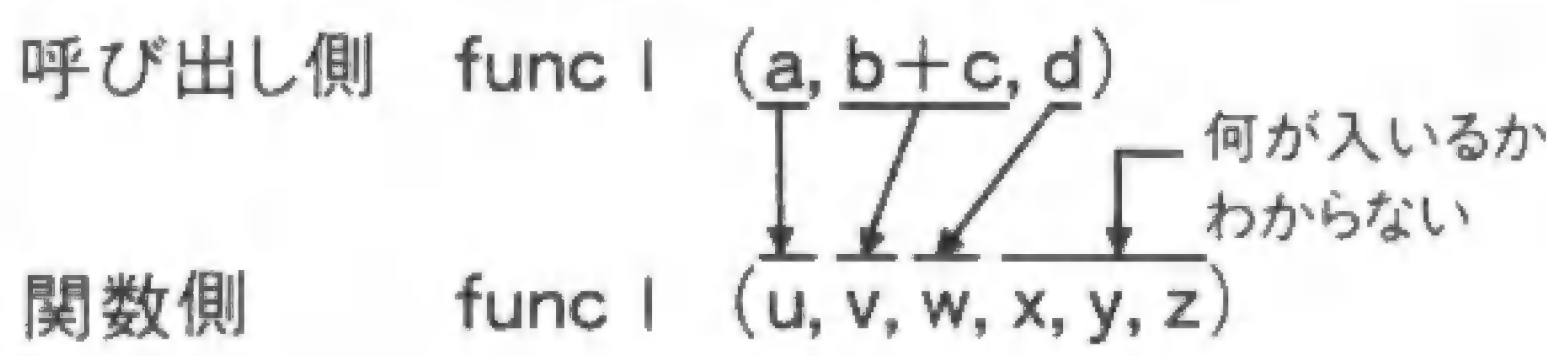
### (6) 関数の呼び出し

実引数には 8 ビット長と 16 ビット長の二種類があり，実引数の値が関数に渡されます。セミコロン (；) により前が 8 ビット長の実引数で，0 ～32 個の式が書けます。セミコロンより後が 16 ビット長の実引数で，インデックス・レジスタ名や定数式を書きます。関数の呼び出しで書かれる実引数は次のような意味を持っています。

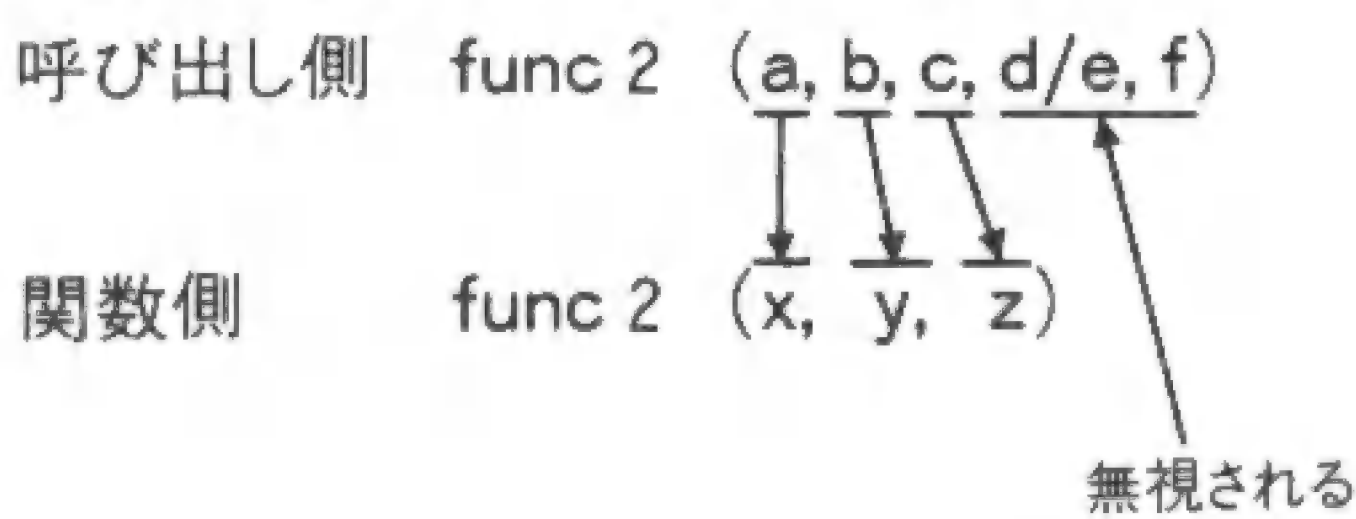


つづき

●仮引数の数>実引数の数の場合



●仮引数の数<実引数の場合



(7) 関数の値の渡しかた

関数の値は return 文の一つ前に実行した式の値となります。

たとえば仮引数 a, b, c の合計を関数の値とする関数は、

```
add 3 (a, b, c) ;  
{  
    a + b + c ;  
}
```

となります。次の例は、ix が示す n バイトのメモリの中から x と等しいものを探し出す関数で、等しいものがあればその位置を、等しいものがない場合は \$FF を関数と値として戻します。

```
sear (n, x ; ix) ;  
var i ;  
{  
    if n = 0 then {  
        $ff ;  
        return ;  
    }
```

つづく



つづき

```
for i := 0 to n - 1 {  
    if memory[ix+] = x  
    then {  
        i;  
        return;  
    }  
}  
$ff;  
}
```

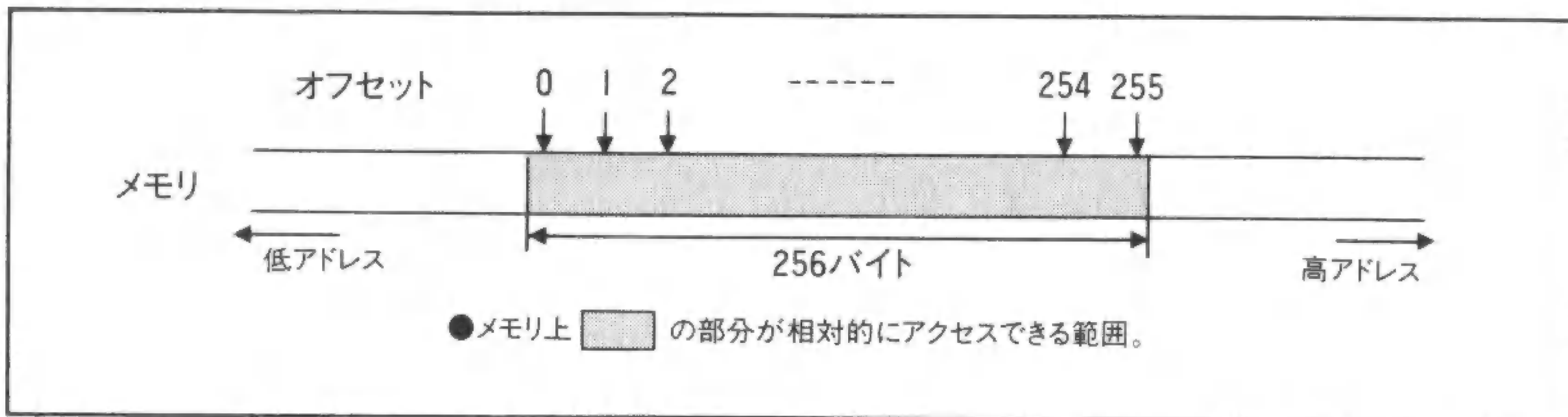
## [5] 変数と定数

### (1) 変数

変数として扱うものには、変数宣言部で宣言した変数名、データ宣言部で宣言したデータ名、そして memory 配列と port 配列です。データ名は： $=$ の左辺に置いて値を代入することはできません。

変数名、データ名の後には [式] をつけて相対的な参照、代入（代入は変数名のみ）が行なえます。[] 内の式が変数あるいはデータからのオフセットで 0～255 までの値となります。図 4-15 が相対的にアクセスできる範囲を表わした図です。

図4-15 相対的にアクセスできる範囲



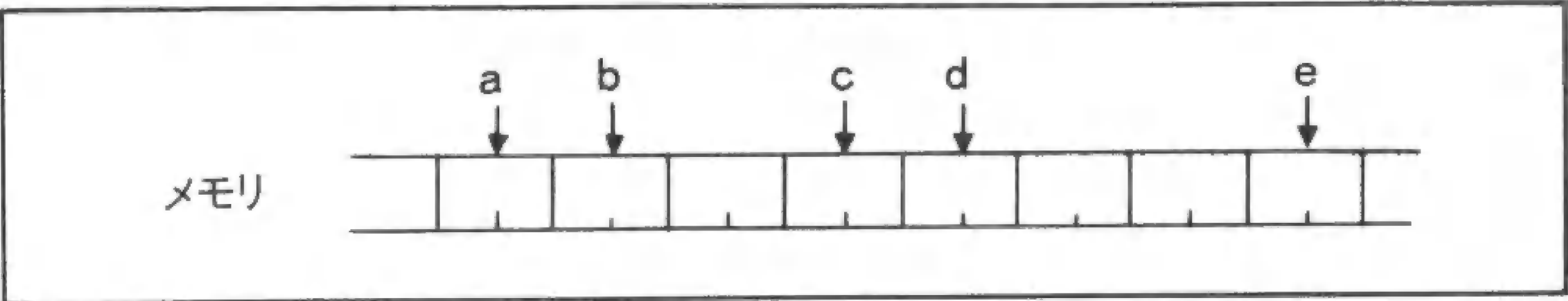
```
var a, b [2], c, d [3], e;  
{  
    b := d [1] := $ 1 d;
```

つづく

つづき

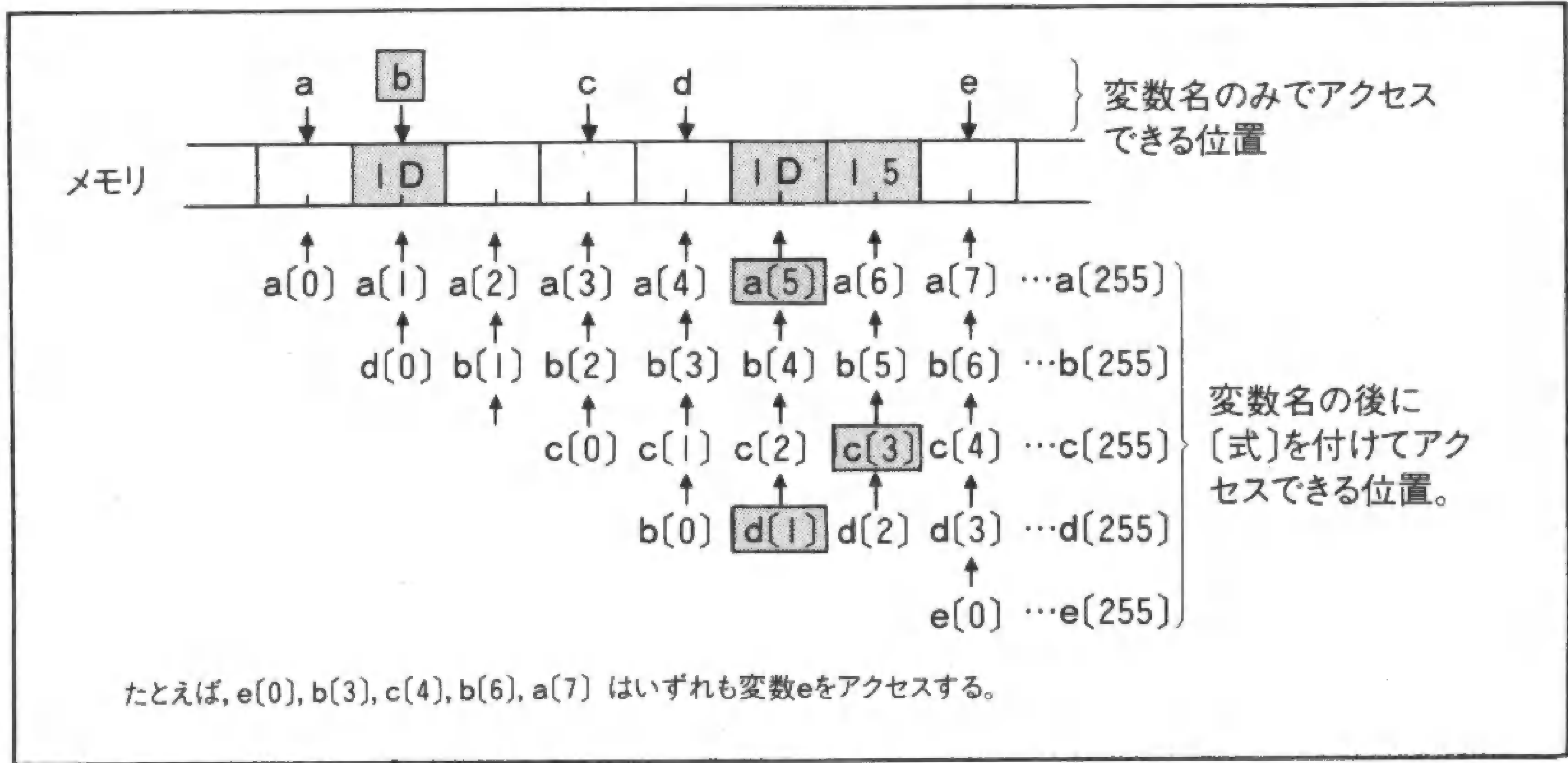
```
c [ 3 ] := a [ 5 ] - 8 ;  
}
```

この例の場合、変数宣言部では変数はメモリ上に次のように割りつけられます。



a, c, eはそれぞれ1バイト、bはb[2]と宣言されているので2バイト、dはd[3]と宣言されているので3バイト取られます。

この例を実行するとメモリ上の値は次のようになります。



memory 配列は CPU の全メモリ空間 64 Kバイトに対して自由に参照、代入するためのもので、二つの形式があります。

memory [式1, 式2]

式1の値を上位バイトとし、式2の値を下位バイトにしてメモリのアドレスを設定し参照、代入を行ないます。



memory [インデックス・レジスタ名, 定数]

この形式はインデックス・レジスタ (ix, iy) を使うものです。基本的な使用法は上記のように、初めにインデックス・レジスタ名 (ix, iy) を書き、次に定数を書きます。

インデックス・レジスタ名の後に正符号 (+) または負符号 (-) をつけると参照、代入を行なった後、インデックス・レジスタの値を正符号 (+) で +1, 負符号 (-) で -1 します。

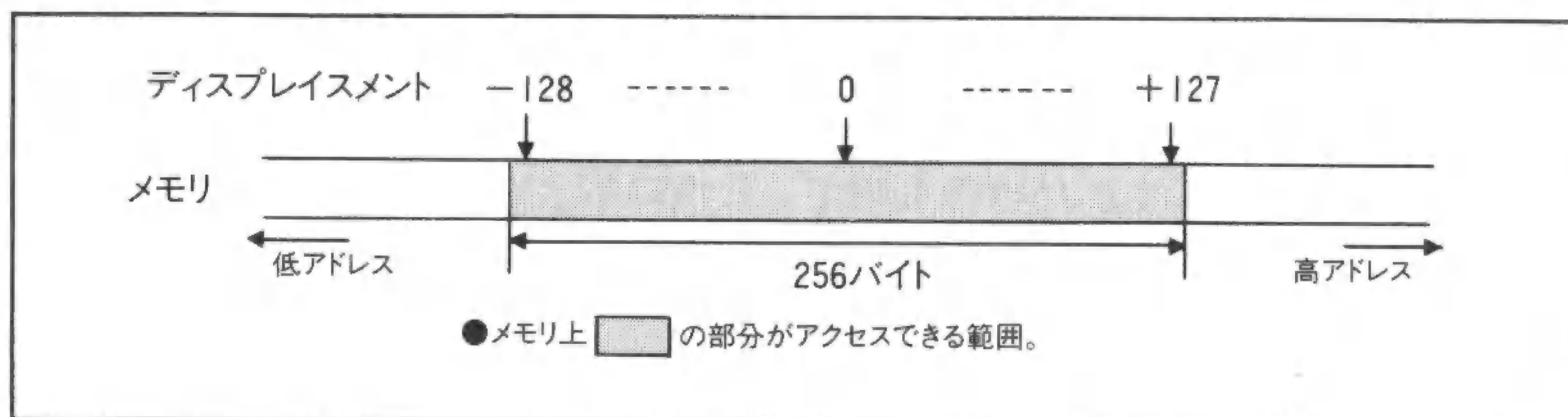
定数は符号をつけないで場合で 0 ~ 127 までの整数, 符号をつけた場合で -128 ~ +127 までの整数となります。定数の値がゼロの場合,

memory [インデックス・レジスタ名]

と書いてもかまいません。この定数のことをディスプレイメントといい、この定数により、インデックス・レジスタが示すアドレスを中心に -128 ~ +127 バイトがアクセスできるようになります (図 4-16)。また、“memory”の代わりに @ と書くことができます。

port 配列は、I / O ポートに対して参照、代入を行なうものです。しかし、これは“out (n), a”, “in a, (n)” というオブジェクト・コードに展開されるので、bc レジスタを使った 16 ビットのポート・アドレス指定はできません。“port”の代わりに @@ と書くこともできます。

図4-16 ディスプレイメントとアクセスできる範囲



## (2) 定数

定数としては次のようなものがあります。

- 10 進数 0 ～65535 までの正の整数。
- 16 進数 \$ を頭につけた 0 ～FFFF までの 16 進数。
- 文字 1 文字を引用符 (') で囲んだもの。
- 定数名 定数定義部で定義したとき、値につけた名前。
- アドレス 定数名、データ名、ラベル名、関数名の頭にピリオド (.) をつけるとそのアドレスが定数となります。
- 表意定数 コンパイラが自動的に定義する大域的な定数名のことです。表意定数には次の三つのものがあります。

\_work…ランタイム・ルーチンのワーク・エリアの先頭アドレス。この内容は関数の実引数の数の数。

\_var …変数エリアの先頭アドレス。

\_code …オブジェクト・プログラムの先頭アドレス。

- ロケーション・カウンタの値 \$ のあとが 16 進数以外の文字なら、その \$ は次の生成するオブジェクト・コードのアドレスを示します。

定数は基本的には 16 ビットの値を取ります。ただし、式などで使う場合、定数は 8 ビット長の値 (10 進数で 0 ～255, 16 進数で 0 ～FF) でなければなりません。

もし 9 ビット以上の値 (10 進数で 256～65535, 16 進数では 100～FFFF) だとエラーになります。そのような場合、定数の前に hi または low と書けば上位または下位の 8 ビットだけを定数とすることができます。

たとえば、変数 abc のアドレスが 16 進数で A25C のとき、low. abc とするとこの値は 16 進数で 5C となり、hi. abc では 16 進数で A2 となります。



## 4-4-2 Stellarエラーメッセージ

コンパイル中に表示されるエラーメッセージは、次のとおりです。

1. 行番号：Missing variable name：変数名

内容：指定された変数名が見つからない。

動作：アドレスがゼロの全域的な変数として処理を続ける。

2. 行番号：Missing constant name：定数名

内容：指定された定数名が見つからない。

動作：値がゼロの全域的な定数名として処理を続ける。

3. 行番号：Bad option switch

内容：オプション・スイッチの指定が悪い。

動作：アボート

4. 行番号：Illegal function name：名前

内容：不法な関数名で関数を宣言しようとした。

動作：名前が予約語や定数ならアボート。二重宣言ならその名前でシンボル・テーブル（記号表）へ再登録し処理を続ける。

5. 行番号：Illigal name：名前

内容：不法な名前で定数名、変数名を宣言しようとした。

動作：名前が予約語や定数ならアボート。二重定義、宣言ならその名前でシンボル・テーブルへ再登録し処理を続ける。

6. 行番号：Illingal label：ラベル

内容：不法な名前でラベルを定義しようとした（二重定義など）。

動作：その名前でシンボル・テーブルへ登録される。

## 7. 行番号: Bad string data

内容: 文字列定数の指定がない (文字列の右の"がないまま行が終わっている)。

動作: 文字列定数の右に二重引用符 (") があるものとして処理を続ける。

## 8. 行番号: Too many arguments

内容: 実引数の数が 32 個以上定義されている。

動作: 処理を続ける。

## 9. 行番号: Illigal character: 文字

内容: 字句として認められない文字がある。

動作: アボート

## 10. ? : Undefined label: ラベル

内容: 未定義なラベルがある。

動作: 処理を続ける。

## 11. ? : Undefined function name: 関数名

内容: 未定義な関数名がある。

動作: 処理を続ける。

## 12. 行番号: Illigal constant: 定数

// displacement

// over range

内容: 不法な定数が指定された。定数値, displacemet, over range のいずれかが表示される。

動作: 値をゼロにして処理を続ける。

## 13. 行番号: Bad constant

内容: 定数の指定が悪い。

動作: アボート

## 14. 行番号: Bad address constant

内容: アドレスの指定が悪い。または指定ができないところでアドレス定数を指定した。

動作: アボート

## 15. 行番号: Syntax error

内容: 構文が正しくない。その他のエラー。

動作: アボート



16. 行番号：Bad index operation

内容：インデックス操作文が正しくない。

動作：アボート

17. 行番号：Bad expression

内容：式が正しくない。

動作：アボート

18. %Abort

内容：アボート

動作：コンパイルを異常終了させる。

19. 行番号：Bad include file name

内容：%includeで指定されたファイル名が正しくない。

動作：アボート

20. 行番号：No include file

内容：%includeで指定されたファイルがない。

動作：アボート

21. %Include source is not on memory

内容：%includeで指定されたファイルがメモリ上にない。

動作：アボート

22. %Symbole table overflow

内容：シンボル・テーブルがオーバーフローした。

動作：コンパイルを中止する。

23. %Object area full

内容：オブジェクト・プログラムを出力する領域がいっぱいになった。

動作：コンパイルを中止する。

24. %Bad source file

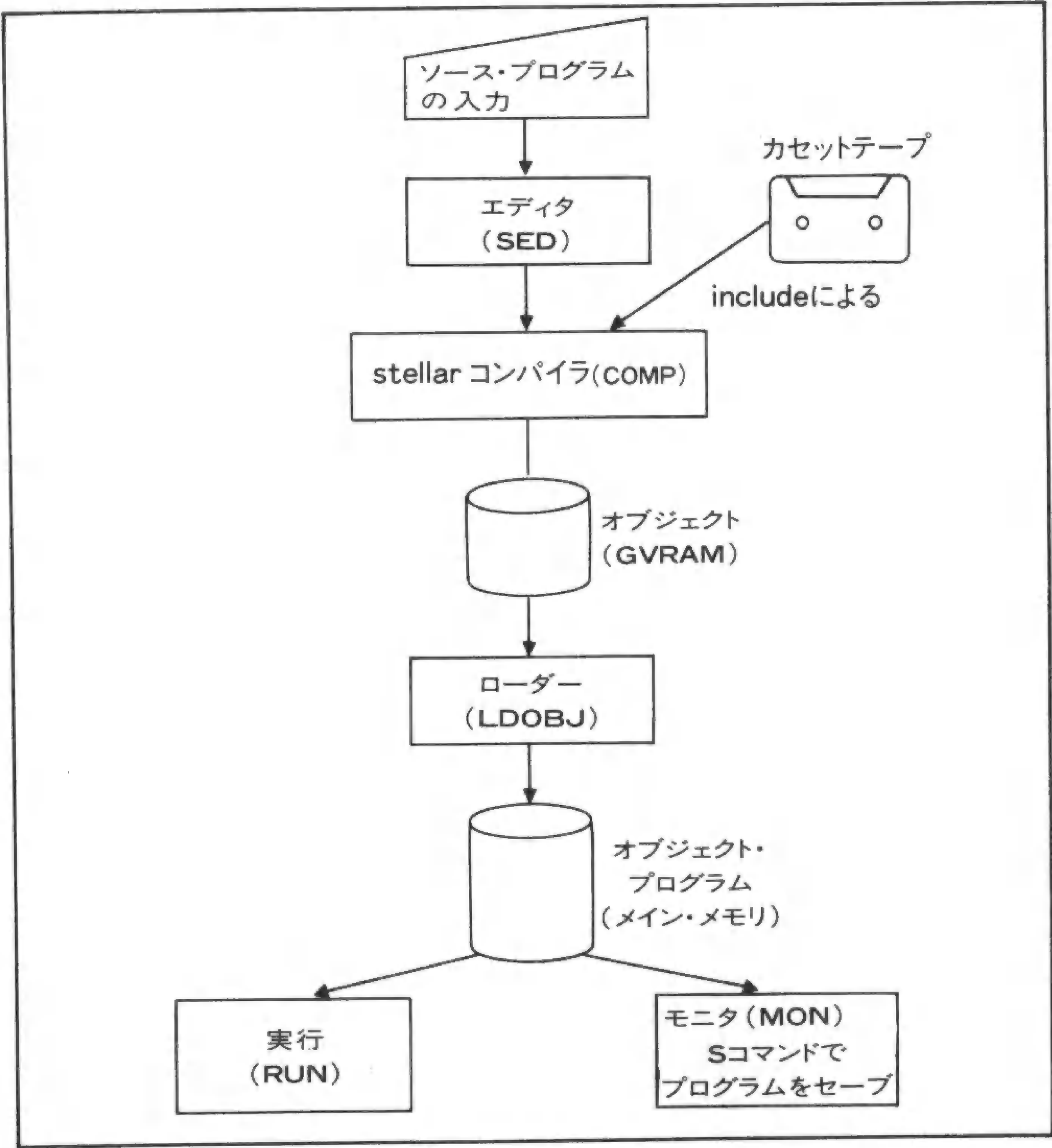
内容：入力したソース・プログラムが正しくない。

動作：コンパイルを中止する。

# 4-4-3 コンパイラの使い方

図 4-17 は Stellar コンパイラでのプログラム開発手順を示したものです。Stellar のソース・プログラムはエディタで作成します。Stellar コンパイラはソース・プログラムを入力してオブジェクト・コードを GVRAM 上に出力します。コンパイラは直接マシン語のオブジェクト・コードを出力しますが、特殊なコードを含んでいるため、このままでは実行することはできません。そこで、これを実行可能にするためのプログラムがローダーで GVRAM 上のオブジェクト・コードを変換してメイン・メモリにロードします。ロードされたオブジェクト・プログラムは実行コマンドで実行できます。

図4-17 Stellarコンパイラでのプログラム開発手順





ここではハノイの塔（リスト 4-1）のプログラムを例にと  
って説明します。

リスト4-1

```

/*      tower of hanoi      */

prog hanoi();
cons    cr := 13, on := 1, off := 0;
var     n;
data    msg1:  "tower of hanoi", cr, 0,
        msg2:  "N ( 1...9 ) = ? ", 0;
{
    lpt(on);
    putstr(;;msg1);
    {
        newline(); putstr(;;msg2);
    } until (n:=getchr()) >= '1' & n <= '9';
    n := n - '0';
    newline();
    move(n, 'X', 'Y', 'Z');
    lpt(off);
}

lpt(sw);
var     _lpt    at ($1472);
{
    _lpt := sw;
}

putstr(;;ix);
{
    inline    $dd,$e5,          /* push ix      */
              $dl,             /* pop  de      */
              $cd,$142f;        /* call printp  */
}

newline();
{
    inline    $cd,$1446;        /* call criprp  */
}

getchr();
{
    inline    $3e,$01,          /* ld   a,1     */
              $cd,$001b,        /* call inkey   */
              $cd,$1420;        /* call accprp  */
}

putchr(c);
{
    inline    $3a,#.c,          /* ld   a,(.c)  */
              $cd,$1420;        /* call accprp  */
}

recursive move(n,x,y,z);
data    mvms:  "move ", 0,
        frms:  " from ", 0,
        toms:  " to ", 0;
{
    if n > 1 then {
        move(n-1,x,z,y);
        putstr(;;mvms);
        putstr(;;frms);
        putstr(;;toms);
        newline();
        move(n-1,z,y,x);
        putchr(n+'0');
        putchr(x);
        putchr(y);
    }
}

```

つづく



リスト4-1 つづき

```

    }else {
        putstr(;.mvms);      putchar(n+'0');
        putstr(;.frms);      putchar(x);
        putstr(;.toms);      putchar(y);
        newline();
    }
}

```

## [1] Stellar コンパイラ

Stellar コンパイラはコマンド・モードのときに“COMP [ /P ] ”と入力して起動します。[ ] で囲まれているものは省略できますが，“/P”を付けるとコンパイル中のメッセージをプリンタにプリントします。また、F2（ファンクションキーの2番）を押すと“COMP ”と入力した場合と同じです。

コンパイル中にエラーがあるとエラーメッセージ、エラー行、そして

<space> continue, E (dit)

と表示して入力待ちになります。スペース・キーを押すとコンパイルは続行し、E（またはe）を押すとエディタに入りエラー箇所の付近にカーソルが移動するので再編集してください。

Stellar コンパイラはエラーがなければ次のようなメッセージを表示してコンパイルします。

```

Stellar compiler Rev 1.00
( X1 Version )
Copyright (c) 1984 H.Watanabe & H.Ohnuki / MIA

Program name : hanoi .....プログラム名
Function name : lpt
Function name : putstr
Function name : newline
Function name : getchr   } 関数名
Function name : putchar
Function name : move

Program 0324 (B000-B323).....生成したオブジェクト・コードのサイズとアドレス
Data    0037 (D900-D936).....変数, ワーク・エリアのサイズとアドレス
Stack bottom ( -F8FF).....スタックのアドレス

** End of compile, No error(s)

```



## [2] ローダー

ローダーはコマンド・モードのときに“LDOBJ [/P] [F3]”と入力して起動します。“/P”を付けるとロード中のメッセージをプリンタにプリントします。また、F3を押すと“LDOBJ [F3]”と入力した場合と同じです。

プログラム中でオプション・スイッチを使っている場合でロード・アドレスがシステム領域と重なっているときは、そのアドレスと“Load error”と表示し、ロードは中断します。ソース・プログラムと重なっているときは

Destroy a File (Y/N) ?

と表示してソース・プログラムを破壊するかどうか聞いてきます。ここでY（またはy）を押すとソース・プログラムを破壊してオブジェクト・コードをロードします。N（またはn）を押すとロードを中断しエラーを表示します。

ロードの一時停止はスペース・キーを押してください。一時停止の解除もスペース・キーです。また、SHIFT+ BREAKキーか ^ Cを押すとロードは中断しエラーを表示します。

ローダーはエラーがなければ次のようなメッセージを表示してオブジェクト・コードをロードします。

Load stellar compiler object

Program name : hanoi = B000	.....プログラムの開始アドレス
Function name : lpt = B19B	} 関数の開始アドレス
Function name : putstr = B1B5	
Function name : newline = B1D6	
Function name : getchr = B1ED	
Function name : putchr = B209	
Function name : move = B223	} 表意定数の値
Constant name : _work = D900	
Constant name : _var = D930	
Constant name : _code = B000	} 全域的な定数名の値
Constant name : cr = 000D	
Constant name : on = 0001	
Constant name : off = 0000	} 全域的な変数のアドレス
Variable name : n = D930	
Data name : msg1 = B10A	} 全域的なデータ名のアドレス
Data name : msg2 = B11A	
End address : B323	.....生成したオブジェクト・コードのエンド・アドレス
Program size : 0324	.....生成したオブジェクト・コードのサイズ

### [3] オブジェクト・プログラムの実行

ロードされたオブジェクト・プログラムの実行はコマンド・モードのときに“RUN [ &Hnnnn] ␣” (nnnn は 16 進数) と入力すると実行できます。また、F 5 を押すと“RUN ␣”, F 10 を押すと“RUN &H”と入力した場合と同じです。“&Hnnnn”を付けると nnnn 番地から実行できます。これはロード中に表示された関数のデバックなどに便利です。

オブジェクト・プログラムがデバック・モードで %tron が指定されているとき、実行を一時停止するときはスペース・キーを押してください。一時停止の解除はスペース・キーです。また、実行を中断するには SHIFT+BREAK キーか ^C を押してください。

実行結果は次のようになります。

```
tower of hanoi
N ( 1...9 )= ? 4
move 1 from X to Z
move 2 from X to Y
move 1 from Z to Y
move 3 from X to Z
move 1 from Y to X
move 2 from Y to Z
move 1 from X to Z
move 4 from X to Y
move 1 from Z to Y
move 2 from Z to X
move 1 from Y to X
move 3 from Z to Y
move 1 from X to Z
move 2 from X to Y
move 1 from Z to Y
```

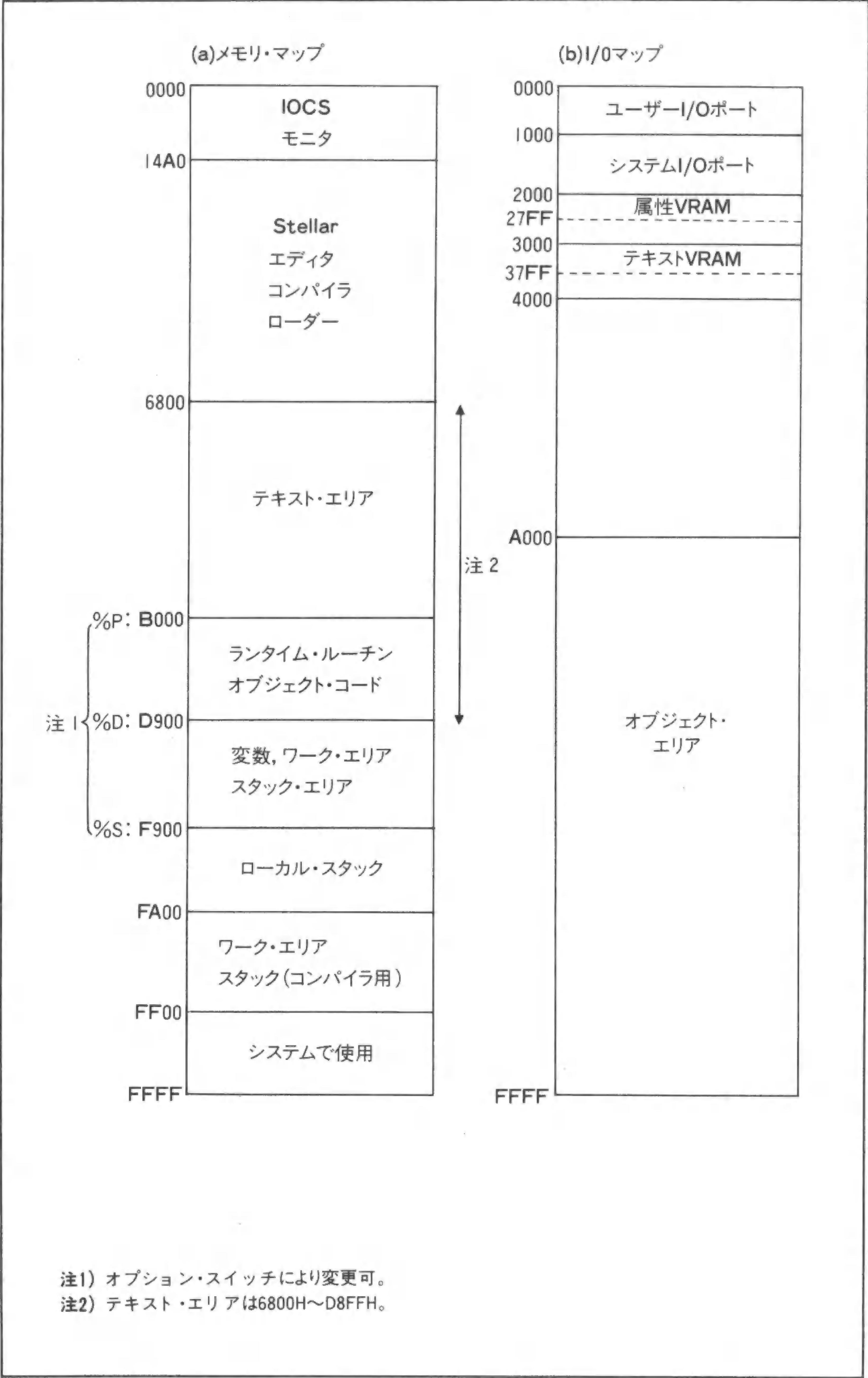
### [4] オブジェクト・プログラムの続行

% break で実行を中断したプログラムの再開はコマンド・モードのときに“CONT ␣”と入力するかまたは F 5 を押します。

図 4-18 は Stellar 動作時のメモリ・マップです。



図4-18 Stellar 動作時のメモリ・マップ



## 4-4-4 マシン語とのインターフェイス

Stellar は入出力の文を持っていないので、inline 文を使ってこれをマシン語で記述する必要があります。inline 文を抜けるときはスタック・ポインタを入るときに値に戻さなければなりません。

### [1] マシン語とのインターフェイス

リスト 4-2 は文字列のアドレスを de レジスタにセットして \$000b をコールし、画面に“X 1”という文字列を表示するプログラムです。

リスト 4-3 は‘A’、‘B’という文字を表示し、リスト 4-4 はキーボードから 1 文字入力して表示するプログラムです。Stellar では式や関数の値を a レジスタにセットするのでリスト 4-3、4-4 のような使い方ができます。リスト 4-4 の関数 input の関数値は \$001b をコールしたときにセットされる a レジスタの値です。

リスト 4-5 は 8 ビット・データを関数に渡すプログラムで‘B’という文字を表示します。変数 ch にセットされているのは \$41 ですが、関数 printf に渡されるときは評価 (ch := ch + 1) された値 \$42 が渡されて仮引数 char にセットされます。

16 ビット・データの受け渡しにはリスト 4-6 に示すようにインデックス・レジスタ ix, iy を使います。関数 add は ix := ix + iy という計算をするので、関数の頭書きでは‘#’をつけています。

リスト4-2

```

prog print();
data    string:"X1",0;
{
    inline  $l1,#.string,    /* ld    de..string    */
            $cd,#$000b;      /* call print    */
}

```



リスト4-3

```

prog putc();
{
    $41;
    inline $cd, #0013;    /* call putc */

    $42;
    inline $cd, #0013;    /* call putc */
}

```

リスト4-4

```

prog test();
{
    input();
    inline $cd, #0013;    /* call prt */
}

input();
{
    inline $3e, $01,      /* ld a,1 */
    $cd, #001b;          /* call inkey */
}

```

リスト4-5

```

prog test();
var ch;
{
    ch:=$41;
    printf(ch:=ch+1);
}
printf(char);
{
    inline $3a, #.char,    /* ld a, (.char) */
    $cd, #0013;          /* call putc */
}

```

リスト4-6

```

prog test();
{
    set ix := $1234;
    add(;ix, $1111);
    printx(;ix);
}

add(;#ix, #iy);
{
    inline $dd, $e5,      /* push ix */
    $e1,                /* pop hl */
    $fd, $e5,           /* push iy */
    $d1,                /* pop de */
    $19,                /* add hl, de */
    $e5,                /* push hl */
    $dd, $e1;           /* pop ix */
}

```

つづく

リスト4-6 つづき

```

}

printx(;ix);
{
    inline $dd,$e5,      /* push ix      */
          $e1,          /* pop  hl      */
          $cd,#$1202;    /* call hlhxr   */
}

```

## [2] I/Oポートへのアクセス

StellarではX 1の16ビットのI/O空間をアクセスできないので、リスト4-7に示すような関数outport, inportを使います。

リスト4-7

```

prog inout();
{
    outport($42;$3000);
    inport($3000);
    inline $cd,$0013;    /* call putc   */
}

outport(v;$ix);
{
    inline $dd,$e5,      /* push ix      */
          $c1,          /* pop  bc      */
          $3a,#.v,      /* ld   a,(.v)  */
          $ed,$79;      /* out  (c),a   */
}

inport($ix);
{
    inline $dd,$e5,      /* push ix      */
          $c1,          /* pop  bc      */
          $ed,$78;      /* in   a,(c)   */
}

```

## [3] ランタイム・ルーチン

ランタイム・ルーチンには13個のエントリ・ポイントがあります。各エントリの処理は次のようになっています。

ここで\_codeは表意定数の1つでオブジェクト・コードの先頭アドレスを示しています。

\_code+0：生成したオブジェクト・プログラムを実行する。

\_code+3：オブジェクト・プログラムの実行を終わらせる。

\_code+6：1バイトの乗算， $a \leftarrow a * (hl)$ 。de, hlは破壊される。



- \_code+9 : 1 バイトの乗算,  $a \leftarrow a * d$ 。de, hl は破壊される。
- \_code+12 : 1 バイトの除算,  $a \leftarrow a / (hl)$ 。de, hl は破壊される。
- \_code+15 : 1 バイトの除算,  $a \leftarrow a / d$ 。d, hl は破壊される。
- \_code+18 : 1 バイトの剰余,  $a \leftarrow a \% (hl)$ 。de, hl は破壊される。
- \_code+21 : 1 バイトの剰余,  $a \leftarrow a \% d$ 。d, hl は破壊される。
- \_code+24 : 左シフト。  $a \leftarrow a << (hl)$ 。de は破壊される。
- \_code+27 : 左シフト。  $a \leftarrow a << d$ 。de は破壊される。
- \_code+30 : 右シフト。  $a \leftarrow a >> (hl)$ 。de は破壊される。
- \_code+33 : 右シフト。  $a \leftarrow a >> d$ 。de は破壊される。
- \_code+36 : 関数への実引数を仮引数へ転送する。また、再帰的な関数の場合は局所的な変数のセーブも行なう。bc, de, hl, a は破壊される。

また、ランタイム・ルーチンはワーク・エリアとして 48 バイトの RAM 領域を使用しています。そのうち、次の 2 つの領域は Stellar のプログラムで参照可能な領域です。ここで \_work は表意定数の 1 つでワーク・エリアの先頭のアドレスを示しています。

- \_work + 0 : 実引数の数。
  - \_work + 1 ~ 32 : 実引数の値 (第 1 引数 ~ 第 32 引数まで)。
- リスト 4-8 の print という関数は、与えられた引数の個数分だけ表示するものです。

リスト4-8

```

prog test();
{
    print('a','b','c','d');
}

print();
var    c[33],z at(_work);
{
    set ix:=.c;
    set iy:=.z;
    loop #,z+1 {
        @[ix+]:=@[iy+];
    }
    set ix:=.c+1;
    loop #,c {
        @[ix+];
        inline $cd,$$0013; /* call putc */
    }
}

```

## 4-4-5 ライブラリの使い方

ライブラリには、基本ライブラリ、スクリーン制御ライブラリ、コンソール入出力ライブラリ、算術ライブラリがあります。

### [1] 基本ライブラリ

基本ライブラリはサブルーチンをコールするものです。レジスタの設定、参照は代入文でできます（リスト4-9）。WORKBUFはコンソール入出力ライブラリの文字列の入力で使われるだけなので自由に使用してください。

#### (I) call（；アドレス）

アドレスで指定された番地をコールします。代入文でレジスタの値をセットしてください。

リスト4-9

```

/*****
* 1      X1 Basic Library                      *
*      Rev 1.0   1st November 1984            *
*****/

var    _af[2],_bc[2],_de[2],_hl[2],_ix[2],_iy[2],
        _a at(._af+1),_b at(._bc+1),
        _d at(._de+1),_h at(._hl+1);

```

つづく



リスト4-9 つづき

```

var      WORKBUF[256];

        /* X1 IOCS call */

call(;ix);
{
    inline $21, #.rt,      /* ld    hl, .rt    */
            $e5,          /* push hl        */
            $dd, $e5,     /* push ix        */
            $2a, #._af,   /* ld    hl, (.af) */
            $e5,          /* push hl        */
            $f1,          /* pop  af        */
            $ed, $4b, #._bc, /* ld    bc, (._bc) */
            $ed, $5b, #._de, /* ld    de, (._de) */
            $2a, #._hl,   /* ld    hl, (._hl) */
            $dd, $2a, #._ix, /* ld    ix, (._ix) */
            $fd, $2a, #._iy, /* ld    iy, (._iy) */
            $c9;          /* ret           */

    rt:
        inline $fd, $22, #._iy, /* ld    (._iy), iy */
                $dd, $22, #._ix, /* ld    (._ix), ix */
                $22, #._hl,   /* ld    (._hl), hl */
                $ed, $53, #._de, /* ld    (._de), de */
                $ed, $43, #._bc, /* ld    (._bc), bc */
                $f5,          /* push af         */
                $e1,          /* pop  hl         */
                $22, #._af;    /* ld    (._af), hl */
}

```

## [ 2 ] スクリーン制御ライブラリ

スクリーン制御ライブラリの関数の機能は HuBASIC とほぼ同じです (リスト 4-10)。

(I) width (<カラム数>)

(II) console (ys, yl, xs, xl)

パラメータをすべて省略すると、最大文字数表示をする画面に戻ります。

(III) sereen (<出力ページ>, <入力ページ>, gmode)

gmode が 1 のときグラフィックを表示し、0 のときは表示しません (1 行あたりのカラム数が 40 のときのみ有効)。

(IV) color (<カラーコード>)

文字の色を指定する命令です。

(V) bgcolor (<カラーコード>)

背景の色を指定する命令です。

(VI) cls (n)

n の値は次のとおりです。

0…グラフィック 1, グラフィック 2, グラフィック

3を同時にクリアします。

1…グラフィック1をクリアします。

2…グラフィック2をクリアします。

3…グラフィック3をクリアします。

4…グラフィック1, グラフィック2, グラフィック3とテキスト画面を同時にクリアします。

5…テキスト画面をクリアします。

(VII) locate (x, y)

リスト4-10

```

/*****
*
* 2   Screen Control Library
*     28th Octber 1984 by H.Watanabe
*
*     2.1 width(wsize)
*     2.2 console(ys,yl,xs,xl)
*     2.3 screen(outpage,inpage,gmode)
*     2.4 color(color_code)
*     2.5 back_color(color_code)
*     2.6 cls(n)
*     2.7 locate(x,y)
*
*****/

var      LINLIM   at($6) ,      WIDTH0   at($7),
         CURX     at($E) ,      CURY     at($F),
         CURYST   at($16) ,     CURYED   at($17),
         CURXST   at($1E) ,     CURXED   at($1F),
         COLORF   at($26) ,     CLSCHR   at($27);

width(wsize);
{
    _a := wsize;
    call(;$988);
}

console(ys,yl,xs,xl);
{
    if ( ys = 0 & yl = 0 & xs = 0 & xl = 0 ) then
    {
        yl := 25;
        xl := WIDTH0;
        goto OK;
    }

    if (ys < 0 | ys > 24) then goto ERROR;
    if (yl < 1 | yl > 25-ys) then goto ERROR;
    if (xs < 0 | xs > WIDTH0-1) then goto ERROR;
    if (xl < 1 | xl > WIDTH0-xs) then goto ERROR;

OK:
    CURYST := ys;   CURYST := ys + yl;
    CURXST := xs;   CURXED := xs + xl;

    i:      return; /* normal return */
ERROR:
    0:      return; /* error return */
}

```

つづく



リスト4-10 つづき

```

screen(outpage,inpage,gmode);
{
    _a := outpage;
    call(;$9c0);
    _a := inpage;
    call(;$9f5);
    if gmode then call(;$a5a);

    else inline
        $01,$1000,    /* ld bc,1000h */
        $af,          /* xor a */
        $ed,$79,      /* out (c),a blue */
        $04,          /* inc b */
        $ed,$79,      /* out (c),a red */
        $04,          /* inc b */
        $ed,$79,      /* out (c),a green */
        $04,          /* inc b */
        $ed,$79;      /* out (c),a priority */

}

color(color_code);
{
    color_code := color_code % 8;
    COLORF := (COLORF & $f0) | color_code;
}

back_color(color_code);
{
    if color_code > 7 then color_code := color_code % 8;
    inline
        $3a, $.color_code, /* ld a, (.color_code) */
        $21, $f6, 0,      /* ld hl, RPRIOF */
        $06, 3,          /* ld b, 3 */
        $cb, $1e,        /* loop rr (hl) */
        $0f,             /* rrca */
        $cb, $16,        /* rl (hl) */
        $23,             /* inc hl */
        $10, $f8,        /* djnz loop */
        $cd, $5a, $0a;    /* call stprio */
}

cls(n);
/*      n..0      Graphic Clear
        1      Vram(4000H-7FFFH) Clear
        2      Vram(8000H-BFFFH) Clear
        3      Vram(C000H-FFFFH) Clear
        4      Text Clear Graphic(4000H-FFFFH)
        5      text Clear
*/
{
    if ( n = 0 ) then call(;$a8f); /* Graphic Clear */
    elseif (n = 1) then fillvram(0;$4000,$4000);
    elseif (n = 2) then fillvram(0;$8000,$4000);
    elseif (n = 3) then fillvram(0;$c000,$4000);
    elseif (n = 4) then
        {
            call(;$6e4); /* text clear & cursor home */
            call(;$a8f); /* Graphic clear */
        }
    elseif (n = 5 ) then call(;$6e4); /* text clear */
    else {0; return;} /* error */

    l; /* normal return */
}

locate(x,y);

```

リスト4-10 つづき

```

{
    if (y < CURYST ! y > CURYED ) then goto ERROR;
    if (x >= CURXST & x < CURXED ) then
    {
        CURX := x;
        CURY := y;
        l;
        return;
    }
    ERROR:
    0; return;
}

fillvram(c;ix,iy);
/*      c      fill chracter
   ix      fill start address
   iy      fill length
*/
{
    inline
    $dd,$e5,      /*      push      ix      */
    $c1,          /*      pop       bc      */
    $fd,$e5,      /*      push      iy      */
    $e1,          /*      pop       hl      */
    $3a,#.c,      /*      ld        a,(c)   */
    $57,          /*      ld        d,a     */
    $ed,$51,      /* loop:out      (c),d    */
    $03,          /*      inc       bc      */
    $2b,          /*      dec       hl      */
    $7c,          /*      ld        a,h     */
    $b5,          /*      or        l       */
    $20,$f8;      /*      jr        nz,loop */
}

```

### [ 3 ] コンソール入出力ライブラリ

コンソール (CRT, キーボード) との入出力やプリンタへの出力のための関数です (リスト 4-11)。

#### ( I ) print\_str ( c ; <文字列の開始アドレス> )

NUL コード ( 0 ) で終わる文字列を出力します。文字列を出力した後, C が 1 のときは改行し, 0 のときは何もしません。

#### ( II ) print\_num ( <数> )

符号なしの 1 バイトの数を 10 進数で出力します。

#### ( III ) print\_hex ( <数> )

数を 16 進数で出力します。

#### ( IV ) print\_chr ( <数> )

数をそのまま文字として出力します。

#### ( V ) print\_spc ( <回数> )

回数分だけスペースを出力します。回数を省略したとき (つまり 0 のとき) は 1 回だけ出力します。




(VI) cr(<回数>)

回数分だけ改行します。回数を省略したときは1回改行します。

(VII) input\_str (m, b, s ; <文字列格納アドレス>, <メッセージ先頭アドレス>)

文字列を入力するときに使います。mは入力する文字列の最大長です。bが0のときは1行全体を入力し、0以外のときはメッセージをスキップして文字列を入力します。sが1のときはメッセージを画面に出力します。sを省略した場合は出力しません。この関数値は入力した文字列の長さです(文字列の長さがmより大きいときは-1を返します)。この関数の実行を終了するには次の4通りの方法があります。

①  キー, ^Mによる正常入力。

② ^Jによる現在のカーソルよりも前の文字列の正常入力。

③ SHIFT+BREAK キー, ^Cによる入力キャンセル。

④ ^Dによる入力キャンセル。

入力キャンセルの場合、実行は中断します。

(VIII) input\_num (s ; <文字列格納アドレス>)

キーボードから入力した数値が関数値となります。入力できる数値は10進数で0~255です。sが1のときはメッセージを画面に出力します。sを省略した場合は出力しません。

(IX) input\_chr ( )

キーボードから入力した1文字が関数値となります。inkey (1) と同じです。

(X) inkey (n)

BASIC の inkey \$ と同じです。

(XI) lpt\_sw (sw ; #, #)

文字列の出力をswが0のときにコンソール、1のときにプリンタに切り替えます(“print”で始まる関数名のみ有効)。初期設定では文字列の出力はコンソー

ルとなっています。

(XII) stov ( ; # <文字列のアドレス> )

数字の文字列を数値 ( 0 ~ 255 ) に変換します。

(XIII) vtos ( <文字列の最大長> , <数値> ; <文字列を格納するアドレス> )

数値を 10 進数の文字列に変換します。

(XIV) abort ( s ; <アドレス> )

この関数を実行するとプログラムは停止します。s が 1 のときに<アドレス>からのメッセージと “Abort. . .” と表示します。s を省略するとメッセージは表示されません。

#### リスト4-11

```

/*****
*
* 3      I/O Console Library
*      28th Octber 1984 by H.Watanabe
*
*      3.1  print_str(c;ix)
*      3.2  print_num(v)
*      3.3  print_hex(c)
*      3.4  print_chr(c)
*      3.5  input_str(m,b;ix)
*      3.6  input_num()
*      3.7  input_chr()
*      3.8  inkey(n)
*      3.9  print_spc(n)
*      3.10 cr(n)
*      3.11 lpt_sw(sw;#,#)
*      3.12 stov(;#ix)
*      3.13 vtos(length,value;ix)
*
*****/

var      FILEOUT at($1472);      /* FILEOUT mode */
data     _lpt:0;                 /* init CRT display */

print_str(c;ix);
{
    FILEOUT := _lpt;
    inline  $dd,$e5,             /* push ix      */
            $dl,                 /* pop  de      */
            $cd,$$142f;          /* call lprint  */
    if (c=1) then    cr();
    FILEOUT := 00;
}

print_num(v);
var      bf[5];
{
    set ix:=.bf;
    vtos(5,v;ix);
    while @[ix]=' ' {
        inx ix;
    }
    print_str(;ix);
}

```

つづく



リスト4-11 つづき

```

print_hex(c);
{
    FILEOUT := _lpt;
    _a := c;
    call(;$1207);
    FILEOUT := 00;
}

print_chr(c);
{
    FILEOUT := _lpt;
    _a := c;
    call(;$1420);

    FILEOUT := 00;
}

input_str(m,b,s;ix,iy);
var i;
{
    if (s = 1) then(
        inline
        $fd,$e5,      /* push iy      */
        $dl,          /* pop de       */
        $cd,$0000b;   /* call print  */
    )
    _de := low.WORKBUF;
    _d := hi.WORKBUF;
    if b = 0 then call(;$03); else call(;$15a); /* line input */
    if carry() then stop; /* cancel */
    ldx iy:=_de;
    dec(m);
    for i:=1 to m {
        if (@[ix+1]:=@[iy+1])=0 then {
            i-1; return;
        }
    }
    @[ix]:=0;
    m;
}

input_num(s;ix);
{
    if (s = 1) then(
        inline
        $dd,$e5,      /* push ix      */
        $dl,          /* pop de       */
        $cd,$0000b;   /* call print  */
    )
    _de := low.WORKBUF;
    _d := hi.WORKBUF;
    call(;$15a); /* line input */
    if carry() then stop;
    ldx ix:=_de;
    stov(;;ix);
}

input_chr();
{
    inkey(1);
}

inkey(n);
{
    _a := n;
    call(;$1b);
    _a;
}

```

つづく

## リスト4-11 つづき

```

print_spc(n);
var    z at(_work);
{
    if z=0 then n:=1;

    if n then loop #,n (
        print_chr(' ');
    )
}

cr(n);
var    z at(_work);
{
    if z=0 then n:=1;
    if n then loop #,n (
        print_chr(13);
    )
}

lpt_sw(sw:#,#);
var    LPTFLG at(._lpt);
{
    LPTFLG := ?(sw;1,0);
}

stov(;#ix);
var    c,v;
{
    while @[ix]=' ' {
        inx ix;
    }
    v:=0;
    while (c:=@[ix])>= '0' & c <= '9' {
        v:=c-'0'+v*10;
        inx ix;
    }
    v;
}

vtos(length,value;ix);
{
    loop #,dec(length) {
        @[ix+]:= ' ';
    }
    @[ix-]:=0;
    {
        @[ix-]:=value%10+'0';
        value:=value/10;
    } until value=0 | dec(length)=0;
}

abort(s;ix);
data   mes:"  Abort...",0;
{
    print_chr(7);
    if (s = 1) then print_str(0;ix);
    print_str(1;.mes);
    stop;
}

```

## [4] 算術ライブラリ

算術ライブラリはワード長の加減乗除算の関数と2次元の配列をサポートしています(リスト4-12)。

(I) add\_w (a\_l, a\_h, b\_l, b\_h; <アドレス>)



ワード長の加算。

(II) sub\_w (a\_l, a\_h, b\_l, b\_h; <アドレス>)

ワード長の減算。

(III) mul (a, b)

バイト長の乗算。結果は  $a * b$  の下位バイト。

(IV) mulov ( )

mul 関数の乗算結果の上位バイトの取り出し。

(V) mul\_w (a\_l, a\_h, b\_l, b\_h; <アドレス>)

ワード長の乗算。

(VI) mulov\_w ( ; <アドレス> )

mul\_w 関数の乗算結果の上位バイトの取り出し。

(VII) div (a, b)

バイト長の除算。結果は  $a / b$  の値。

(VIII) mod (a, b)

div 関数の除算の余りを取り出す。

(IX) div\_w (a\_l, a\_h, b\_l, b\_h; <アドレス>)

ワード長の除算。

(X) mod\_w ( ; <アドレス> )

div\_w 関数の除算の余りを取り出す。

(XI) array (x, y; <アドレス>)

2次元配列の配列要素のアドレスを求める。

array (x, y, z; <アドレス>)

3次元配列の配列要素のアドレスを求める。

(XII) load\_array (x, y; <アドレス>)

2次元配列の配列要素の参照。

load\_array (x, y, z; <アドレス>)

3次元配列の配列要素の参照。

(XIII) store\_array (a, x, y, ; <アドレス>)

2次元配列の配列要素へ a の値を代入。

store\_array (a, x, y, z; <アドレス>)

3次元配列の配列要素へ a の値を代入。

このうち add\_w, sub\_w, mul\_w, div\_w はパラメータの渡しかたが同じです。つまり a\_l と a\_h, b\_l と b\_h をそ

れぞれ一つの2バイトの数 (lが下位, hが上位) と見なし, 符号なし2進整数として計算して結果の下位バイトを<アドレス>へ, 上位バイトを<アドレス>+1へ代入します。このとき, add\_w, sub\_w は計算結果に桁上がりや桁下がり (ボロー) が発生していればキャリーフラグを立てます。

mul\_w の計算結果は一般的に4バイト長になりますが, この関数では下位2バイトだけを結果として返します。残る上位2バイトは mulov\_w を呼べば求められます。求めた値の下位バイトは<アドレス>へ, 上位バイトは<アドレス>+1へ代入します。div\_w で発生した余りを mod\_w で求めることができます。求めた余りの下位バイトは<アドレス>へ, 上位バイトは<アドレス>+1へ代入します。

mul, div は1バイト長の計算 (符号なし2進定数) でパラメータの渡しかたは同じです。計算結果は関数の値として返されます。mulov は mul の計算結果の上位バイトを求めるもので, これも結果は関数の値として返されます。

mod は div の除算で発生した余りを求めるもので, 結果は関数の値として返されます。

array, load\_array, store\_array は, Stellar でサポートしていない2次元, 3次元の配列を扱うための関数です。2次元, 3次元の配列は次のように宣言しなければなりません。

#### ● 2次元配列の宣言

```
data <配列名> : <ワークの先頭アドレス>,
<xの長さ>, <yの長さ>, 1 ;
```

#### ● 3次元配列の宣言

```
data <配列名> : <ワークの先頭アドレス>,
<xの長さ>, <yの長さ>, <zの長さ> ;
```

ここで<ワーク>とは, 配列要素を記憶するための領域で, 大きさは (xの長さ) \* (yの長さ) \* (zの長さ) 以上でなければなりません。



array は x, y あるいは x, y, z で示された配列要素 (<配列名> [x, y] や <配列名> [x, y, z]) のアドレスを求める関数で、結果のアドレスはインデックス・レジスタ ix に設定されています。

load\_array は配列要素 (同上) を参照する関数で、参照した値を関数値として返します。

store\_array は配列要素 (同上) へ a の値を代入する関数です。

#### リスト4-12

```

/*****
* 4 arithmetic
*
* 4.1 add_w(a_l,a_h,b_l,b_h;ix,#)
* 4.2 sub_w(a_l,a_h,b_l,b_h;ix,#)
* 4.3 mul(a,b;#,#)
* 4.4 mulov(;;#)
* 4.5 mul_w(a_l,a_h,b_l,b_h;ix,#)
* 4.6 mulov_w(;;ix,#)
* 4.7 div(a_l,a_h,b_l,b_h;ix,#)
* 4.8 mod(;;#)
* 4.9 div_w(a_l,a_h,b_l,b_h;ix,#)
* 4.10 mod_w(;;ix,#)
* 4.11 array(x,y,z;#ix,#)
* 4.12 load_array(x,y,z;ix,#)
* 4.13 store_array(a,x,y,z;ix,#)
*****/

add_w(a_l,a_h,b_l,b_h;ix,#);
{
    inline
    $2a, #.a_l,      /* ld    hl, (.a_l)      */
    $ed, $5b, #.b_l, /* ld    de, (.b_l)     */
    $19,             /* add   hl, de          */
    $dd, $75, $00,   /* ld    (ix), l         */
    $dd, $74, $01;   /* ld    (ix+1), h       */
}

sub_w(a_l,a_h,b_l,b_h;ix,#);
{
    inline
    $2a, #.a_l,      /* ld    hl, (.a_l)      */
    $ed, $5b, #.b_l, /* ld    de, (.b_l)     */
    $b7,             /* or    a               */
    $ed, $52,         /* sbc   hl, de          */
    $dd, $75, $00,   /* ld    (ix), l         */
    $dd, $74, $01;   /* ld    (ix+1), h       */
}

data _mul:00, _mul_w:00,00;

mul(a,b;#,#);
var mul_m at(_code+6);
{
    inline
    $21, #.b,        /* ld    hl, .b          */

```

つづく

## リスト4-12 つづき

```

        $3a, #.a,          /* ld    a, (.a)          */
        $cd, #.mul_m,      /* call mul_m          */
        $af,              /* xor    a            */
        $94,              /* sub    h            */
        $7c,              /* ld     a, h          */
        $32, #._mul,       /* ld     (._mul), a    */
        $7d;              /* ld     a, l          */
    }
mulov(;;#,#);
{
    _mul;
}

mul_w(a_l,a_h,b_l,b_h;ix,#);
{
    inline
    $c5,          /* push bc          */
    $21, #0000,   /* ld    hl, 0000   */
    $ed, $4b, #.a_l, /* ld    bc, (.a_l) */
    $ed, $5b, #.b_l, /* ld    de, (.b_l) */
    $3e, $10,      /* ld     a, 10h     */
    $29,          /* add    hl, hl     */
    $cb, $11,      /* rl     c          */
    $cb, $10,      /* rl     b          */
    $30, $04,      /* jr     nc, s1     */
    $19,          /* add    hl, de     */
    $30, $01,      /* jr     nc, s1     */
    $03,          /* inc    bc         */
    $3d,          /* dec    a          */
    $20, $f2,      /* jr     nz, lp     */
    $ed, $43, #._mul_w, /* ld     (._mul_w), bc */
    $dd, $75, $00,  /* ld     (ix), l     */
    $dd, $74, $01,  /* ld     (ix+1), h   */
    $90,          /* sub    b          */
    $3e, $00,      /* ld     a, 0       */
    $99,          /* sbc    a, c        */
    $c1;          /* pop    bc         */
}
mulov_w(;;ix,#);
{
    @[ix,1] := _mul_w[1];
    @[ix] := _mul_w;
}
data _mod:00, _mod_w:00,00;

div(a,b;#,#);
var rem_m at(_code+$12);
{
    inline
    $3a, #.a,          /* ld    a, (.a)          */
    $21, #.b,          /* ld    hl, .b           */
    $cd, #.rem_m,       /* call .rem_m            */
    $32, #._mod,        /* ld     (._mod), a      */
    $b7,              /* or     a               */
    $7b;              /* ld     a, e            */
}

mod(;;#,#);
{
    _mod;
}

```

つづく



```

div_w(a_l,a_h,b_l,b_h;ix,#);
{
    inline
    $2a,#.a_l,      /* ld    hl, (.a_l)      */
    $ed,$4b,#.b_l,  /* ld    bc, (.b_l)      */
    $11,#0000,      /* ld    de,0000         */
    $3e,$10,        /* ld    a,10h           */
    $29,            /* add   hl,hl           */
    $eb,            /* ex    de,hl           */
    $ed,$6a,        /* adc   hl,hl           */

    $ed,$42,        /* sbc   hl,bc           */
    $13,            /* inc   de              */
    $30,$02,        /* jr    nc,s1           */
    $09,            /* add   hl,bc           */
    $1b,            /* dec   de              */
    $eb,            /* ex    de,hl           */
    $3d,            /* dec   a               */
    $20,$f1,        /* jr    nz,lp           */
    $dd,$75,00,     /* ld    (ix),l          */
    $dd,$74,01,     /* ld    (ix+1),h        */
    $ed,$53,#._mod_w, /* ld    (._mod_w),de    */
    $7b,            /* ld    a,e             */
    $b2;            /* or    d               */
}

mod_w(;ix,#);
{
    @[ix,1]:=_mod_w[1];
    @[ix]:=_mod_w;
}

array(x,y,z;#ix,#);
var adr_l,adr_h;
{
    if @[ix,4]-1 then(
        adr_l:=mul(z,@[ix,3]);adr_h:=mulov();
        inline
        $2a,#.adr_l,  /* ld    hl, (.adr_l)    */
        $3a,#.y,      /* ld    a, (.y)         */
        $5f,          /* ld    e,a             */
        $16,$00,      /* ld    d,00            */
        $19,          /* add   hl,de           */
        $22,#.adr_l;  /* ld    hl, (.adr_l)    */
        mul_w(adr_l,adr_h,@[ix,2],0;.:adr_l);
    )
    else {adr_l:=mul(y,@[ix,2]);adr_h:=mulov();}
    inline
    $2a,#.adr_l,      /* ld    hl, (.adr_l)    */
    $dd,$5e,00,       /* ld    e, (ix)         */
    $dd,$56,01,       /* ld    d, (ix+1)       */
    $19,              /* add   hl,de           */
    $3a,#.x,          /* ld    a, (.x)         */
    $5f,              /* ld    e,a             */
    $16,$00,          /* ld    d,00            */
    $19,              /* add   hl,de           */
    $e5,              /* push  hl              */
    $dd,$e1;          /* pop   ix              */
}

load_array(x,y,z;ix,#);

```

リスト4-12 つづき

```
{
    array(x,y,z;ix);
    @[ix];
}

store_array(a,x,y,z;ix,#);
{
    array(x,y,z;ix);
    @[ix]:=a;
}
```

# 4-4-6 コンパイラの全リスト

Stellarコンパイラの全ダンプ・リストをリスト 4-13 に示します。このとおりに打ち込めば、すぐに Stellar が使えるようになります。

## (1) コンパイラのダンプ・リストと打ち込みかた

ダンプ・リストのチェックサムは 8 バイトのデータの和です。モニタの一部を変更することによりダンプ・コマンド (D コマンド) でチェックサムを表示できます (8 バイト文字は表示されません)。変更箇所を図 4-19 に示すので M コマンドで変更してください。


コンパイラは BASIC と領域が重なっているので、絶対に R コマンドを実行しないでください。実行開始番地は 14 A 0 H なので、“G 14 A 0 

図4-19 モニタの変更箇所

```
:11C8=3A CD 20 14 41 AF 86 23 :D4
:11D0=10 FC CD 07 12 00 00 00 :F2
:11D8=00 00 00 00 00 00 CD 46 :13
```

## (2) IPL 起動

IPL 起動するには M コマンドで次の 2 カ所を変更して、

```
012BH → A0
012CH → 14
```



S コマンドでカセット・テープに

\*S0 67DF 0 : Stellar

としてセーブしてください。

これでセーブされたテープは IPL 起動できるようになります。また、テキスト（ライブラリなど）を Stellar といっしょにセーブして IPL 起動することもあります。テキストの最終アドレスは14B6H, 14B7Hに示してあるのでセーブするときの最終アドレスをテキストの最終アドレスに変更してセーブしてください。たとえば14B6Hが1AH, 14B7Hが97H, なら

\*S0 971A 0 : Stellar

とします。

Stellar が起動すると画面がクリアされて

Source on Memory (Y/N) ?

と表示するので、メモリ上にプログラムがあるときはY（または y）ないときはN（または n）を押してください。この時点で画面に“ST>”というプロンプトが表示されまで、コマンド待ちになります（コマンド・モード）。

リスト4-13

:14A0=C3 1C 15 C3 61 15 C3 65 :55	:1550=B6 14 AF 32 B2 FA 18 03 :72
:14A8=16 C3 6F 16 C3 6C 16 00 :A3	:1558=CD 6C 19 21 A0 14 22 2B :74
:14B0=00 68 FF D8 00 00 00 00 :3F	:1560=01 31 00 00 3E 02 32 8B :2F
:14B8=00 17 53 45 44 00 00 4D :40	:1568=0A CD 8C 09 CD 4F 18 CD :6D
:14C0=4F 4E 00 01 43 4F 4D 50 :CD	:1570=A3 04 11 27 16 CD 0B 00 :CD
:14C8=00 02 4C 44 4F 42 4A 00 :6D	:1578=21 A3 14 22 53 10 3E 50 :EB
:14D0=03 52 55 4E 00 04 43 4F :8E	:1580=32 06 00 CD E1 15 21 86 :A2
:14D8=4E 54 00 05 42 4F 4F 54 :DB	:1588=15 E5 CD BA 15 CD A3 04 :0A
:14E0=00 06 41 50 53 53 00 07 :44	:1590=11 23 16 CD 0B 00 11 00 :33
:14E8=46 49 4C 45 53 00 08 00 :7B	:1598=FF CD 03 00 D8 CD EB 15 :74
:14F0=FF A9 14 AB 16 16 3A 27 :F4	:15A0=D8 CD 1A 16 D8 CD F9 15 :88
:14F8=3A 33 3D 94 3D C9 16 03 :5D	:15A8=FE FF C8 87 16 00 5F 21 :E2
:1500=17 16 17 53 6F 75 72 63 :50	:15B0=F1 14 19 7E 23 66 6F C5 :59
:1508=65 20 6F 6E 20 4D 65 6D :A1	:15B8=D1 E9 3E 02 32 8B 0A 21 :E2
:1510=6F 72 79 20 28 59 2F 4E :78	:15C0=61 15 22 7E 14 01 00 10 :3B
:1518=29 20 3F 00 CD E8 39 CD :43	:15C8=AF ED 79 04 ED 79 04 ED :70
:1520=EF 18 AF 32 8B 0A CD 8C :D6	:15D0=79 04 ED 79 04 32 72 14 :9F
:1528=09 11 03 15 CD 0B 00 CD :D7	:15D8=32 1E 00 3E 4F 32 1F 00 :2E
:1530=A3 38 CD 13 00 CD 51 14 :ED	:15E0=C9 AF 32 16 00 3E 17 32 :47
:1538=FE 59 20 1C AF 2A B2 14 :32	:15E8=17 00 C9 21 23 16 EB 1A :3F
:1540=ED 4B B0 14 ED 42 E5 C5 :D5	
:1548=E1 C1 ED B1 20 0A 2B 22 :B7	

つづく



## リスト4-13

つづき

```

:15F0=B7 C8 13 BE 23 37 C0 18 :82
:15F8=F6 11 BA 14 E5 1A 13 B7 :9E
:1600=28 12 47 7E CD 51 14 B8 :E9
:1608=23 28 F2 E1 1A 13 B7 20 :22
:1610=FB 13 18 E8 E5 C1 E1 1A :AF
:1618=C9 23 7E B7 37 C8 FE 20 :3E
:1620=28 F7 C9 53 54 3E 00 58 :25
:1628=31 20 53 74 65 6C 6C 61 :B6
:1630=72 20 63 6F 6D 70 69 6C :16
:1638=65 72 20 56 65 72 20 31 :75
:1640=2E 30 0D 00 7E 23 B7 37 :FA
:1648=C8 FE 20 28 F7 FE 2F 20 :52
:1650=12 7E 23 FE 70 28 04 FE :4B
:1658=50 20 08 3E 01 32 72 14 :6F
:1660=7E B7 C8 37 C9 31 00 00 :2E
:1668=21 86 15 E5 3E 01 01 3E :1F
:1670=00 32 AF 14 CD BA 15 CD :5E
:1678=8C 09 CD B4 18 CD D1 18 :E4
:1680=CD 4F 18 3E 64 32 92 03 :9D
:1688=CD B5 16 3A AF 14 CD 6F :D1
:1690=19 3E 67 32 92 03 CD BF :11
:1698=18 CD 4F 18 3A B8 14 32 :84
:16A0=16 00 32 0F 00 3E 0C CD :6E
:16A8=13 00 C9 CD F4 34 C9 22 :BC
:16B0=0E 00 C3 0B 00 21 F2 0E :FD
:16B8=11 00 01 0E 0A 72 23 06 :C5
:16C0=07 73 23 10 FC 0D 20 F5 :CB
:16C8=C9 11 E6 16 CD 0B 00 CD :7B
:16D0=A3 38 CD 51 14 FE 59 C0 :24
:16D8=11 00 FF 21 FC 16 01 07 :4B
:16E0=00 ED B0 C3 00 FF 0D 41 :AD
:16E8=72 65 20 79 6F 75 20 73 :E7
:16F0=75 72 65 20 28 59 2F 4E :6A
:16F8=29 20 3F 00 3E 1D D3 00 :B6
:1700=C3 00 00 1A FE 2D 28 06 :36
:1708=FE 2B C0 3E 05 01 3E 06 :71
:1710=CD EC 0D C3 C2 2F EB 7E :E3
:1718=B7 C4 44 16 D8 CD A3 04 :21
:1720=21 00 FF 01 20 00 CD 41 :4F
:1728=00 DA C2 2F CD 34 17 CD :B0
:1730=0B 17 18 EC 7E B7 C8 CB :EE
:1738=7F 11 2F 18 20 11 11 2B :44
:1740=18 CB 47 20 0A 11 33 18 :B0
:1748=CB 4F 20 03 11 27 18 CD :5A
:1750=2F 14 CB 76 3E 2A C4 20 :D0
:1758=14 CB 6E 3E 23 C4 20 14 :A6
:1760=CD 3C 14 3E 22 CD 20 14 :7E
:1768=E5 06 0D 23 7E CD 20 14 :9A
:1770=10 F9 3E 2E CD 20 14 06 :7C
:1778=03 23 7E CD 20 14 10 F9 :AE
:1780=11 08 00 19 3E 22 CD 20 :7F
:1788=14 CD 3C 14 3E 27 CD 20 :83
:1790=14 11 00 FF CD BE 17 CD :93
:1798=2F 14 3E 20 CD 20 14 CD :6F
:17A0=F6 17 CD 2F 14 3E 20 CD :48
:17A8=20 14 23 23 23 CD 11 18 :93
:17B0=06 05 1A CD 20 14 13 10 :49
:17B8=F9 CD 46 14 E1 C9 D5 E5 :84
:17C0=7E CD E6 17 3E 2F CD F3 :75
:17C8=17 23 7E 0F 0F 0F 0F E6 :DA
:17D0=0F C6 00 27 CD E6 17 3E :04
:17D8=2F CD F3 17 23 7E CD E6 :5A
:17E0=17 AF 12 E1 D1 C9 F5 0F :57
:17E8=0F 0F 0F CD EF 17 F1 E6 :D7
:17F0=0F C6 30 12 13 C9 E5 D5 :AD
:17F8=23 7E E6 07 5F 16 00 21 :24
:1800=37 18 19 19 19 01 03 00 :9E
:1808=D1 D5 ED B0 AF 12 D1 E1 :B6
:1810=C9 D5 E5 06 02 7E CD E6 :BC
:1818=17 3E 3A CD F3 17 23 10 :99

```

```

:1820=F4 7E CD E6 17 18 BA 41 :4F
:1828=73 63 00 42 69 6E 00 44 :33
:1830=69 72 00 42 61 73 00 53 :44
:1838=75 6E 4D 6F 6E 54 75 65 :3B
:1840=57 65 64 54 68 75 46 72 :09
:1848=69 53 61 74 3F 3F 3F 3A :88
:1850=0F 00 FE 18 20 0C 3D 32 :C0
:1858=0F 00 CD 62 18 3E 0D C3 :64
:1860=13 00 3A B9 14 32 17 00 :63
:1868=3A 17 00 FE 18 D0 01 80 :B8
:1870=27 1E 0A 21 07 00 16 0F :9C
:1878=CD AB 18 3E 07 ED 79 03 :3E
:1880=1D 20 F0 01 80 37 21 42 :48
:1888=0F 1E 0A 16 07 23 7E ED :E2
:1890=79 03 15 20 F8 3E 20 ED :F4
:1898=79 03 C5 01 09 00 09 C1 :15
:18A0=1D 20 E8 01 A7 37 3E 2B :6D
:18A8=ED 79 C9 ED 51 03 2B 7C :17
:18B0=B5 20 F8 C9 E5 D5 C5 21 :36
:18B8=42 0F 11 10 FA 18 09 E5 :72
:18C0=D5 C5 11 42 0F 21 10 FA :27
:18C8=01 A0 00 ED B0 C1 D1 E1 :B1
:18D0=C9 21 42 0F 11 58 19 0E :CB
:18D8=0A 06 02 36 02 23 1A 77 :FE
:18E0=13 23 10 FA 06 0D 36 20 :A9
:18E8=23 10 FB 0D 20 EB C9 21 :30
:18F0=42 0F 11 1A 19 06 0A 22 :C7
:18F8=B0 FA 0E 00 1A 13 B7 28 :C4
:1900=05 23 77 0C 18 F6 E5 2A :C8
:1908=B0 FA 71 E1 3E 0F 91 4F :29
:1910=23 36 20 0D 20 FA 23 10 :D3
:1918=DE C9 53 45 44 0D 00 43 :D3
:1920=4F 4D 50 0D 00 4C 44 4F :D8
:1928=42 4A 0D 00 43 4F 4E 54 :CD
:1930=0D 00 52 55 4E 0D 00 4D :5C
:1938=4F 4E 0D 00 41 50 53 53 :E1
:1940=2B 0D 00 41 50 53 53 2D :9C
:1948=0D 00 46 49 4C 45 53 0D :8D
:1950=00 52 55 4E 20 26 48 00 :83
:1958=0B 42 0B 4B 0B 48 08 07 :05
:1960=11 4D 11 42 11 4B 11 52 :70
:1968=11 43 11 59 C3 A4 28 ED :3A
:1970=73 47 FB 2A 7E 14 22 4B :DE
:1978=FB 21 9E 2D 22 7E 14 31 :CC
:1980=00 FF DD 21 B2 FA F5 CD :6B
:1988=B5 28 CD E6 27 CD F4 27 :9F
:1990=F1 B7 20 06 2A B0 14 22 :DE
:1998=B4 14 CD CB 2D CD 5A 2B :DF
:19A0=DD CB 00 9E CD E6 27 AF :CF
:19A8=01 00 10 ED 79 03 ED 79 :E0
:19B0=03 ED 79 03 ED 79 3C 32 :40
:19B8=3F FB CD 70 2B 21 0A 2B :F8
:19C0=E5 CD 98 1A CD D1 2B CD :FA
:19C8=A3 38 4F FE 20 DA 28 1A :64
:19D0=DD CB 00 4E C2 F7 07 CD :83
:19D8=1D 29 28 22 C5 CD 87 1A :C3
:19E0=C1 CD 06 1A DD CB 00 4E :A4
:19E8=C2 F7 07 CD 83 1A CD D8 :CF
:19F0=28 C8 CD 04 28 2A B4 14 :DB
:19F8=CD 30 28 C3 0D 28 CD F4 :DE
:1A00=1A 0E 0D C3 B4 1D CD D8 :6E
:1A08=28 CA C4 27 DD CB 00 46 :CB
:1A10=C2 6B 27 FE 0D 28 0A 79 :0A
:1A18=FE 0D C8 FE 09 C8 C3 D5 :3A
:1A20=27 79 FE 0D C2 6B 27 C9 :C8
:1A28=CD F4 1A 21 38 1A 87 5F :34
:1A30=16 00 19 7E 23 66 6F E9 :8E
:1A38=82 1A C2 1B 82 1A E0 1C :11
:1A40=26 1B 72 1B F8 1B 28 1D :26
:1A48=45 1B EF 1C 7E 30 DF 23 :1B

```

つづく



リスト4-13 つづき

```

:1A50=04 21 B4 1D 9D 1D 82 1A :4C
:1A58=F7 26 0E 1E BC 1C 45 1B :81
:1A60=69 1D 82 1A FE 1D 51 1C :AA
:1A68=92 1B 48 1D 87 1C 82 1A :51
:1A70=26 1B 45 1B 72 1B 92 1B :DB
:1A78=ED 7B 47 FB 2A 4B FB 22 :3C
:1A80=7E 14 C9 79 C3 13 00 DD :87
:1A88=CB 00 56 C0 CD BA 2A 21 :B3
:1A90=00 08 B7 ED 42 D0 18 12 :E8
:1A98=DD CB 00 4E C8 DD CB 00 :66
:1AA0=56 C8 CD F9 1A DD CB 00 :A6
:1AA8=4E C0 CD E4 2A DD CB 00 :91
:1AB0=4E C0 11 00 01 CD 44 2E :5F
:1AB8=38 01 EB 22 42 FB ED 5B :CB
:1AC0=B4 14 21 55 05 19 22 B4 :32
:1AC8=14 CD A0 27 36 00 ED 53 :1E
:1AD0=B4 14 E5 2A B6 14 22 55 :18
:1AD8=FB 2A B2 14 22 53 FB E1 :3C
:1AE0=22 B6 14 ED 5B 42 FB 19 :8A
:1AE8=22 51 FB 2B 22 B2 14 DD :5E
:1AF0=CB 00 D6 C9 DD CB 00 56 :68
:1AF8=C8 2A B4 14 E5 2A 51 FB :15
:1B00=ED 5B B6 14 B7 ED 52 22 :2A
:1B08=42 FB ED 53 B4 14 2A 55 :C4
:1B10=FB 22 B6 14 2A 53 FB 22 :81
:1B18=B2 14 CD 47 27 DD CB 00 :A9
:1B20=96 E1 22 B4 14 C9 CD D8 :CF
:1B28=28 C8 CD CA 2A FE 0D C2 :7E
:1B30=1C 2A CD 35 29 CD EC 27 :51
:1B38=C0 CD 04 28 2A B4 14 CD :78
:1B40=43 1C C3 0D 28 2A B4 14 :49
:1B48=CD E1 28 D8 CD D7 2A CD :49
:1B50=CF 28 C2 1C 2A CD 40 29 :35
:1B58=28 07 3D 32 0F 00 C3 1C :8C
:1B60=2A CD 72 2A CD 6A 1B C3 :A8
:1B68=1F 2A 3E 0F CD 13 00 C3 :39
:1B70=43 1C CD 72 2A D8 EB CD :58
:1B78=75 2A CD 40 29 28 07 3D :41
:1B80=32 0F 00 C3 E1 29 CD 04 :DF
:1B88=28 CD 6A 1B CD 0D 28 C3 :3F
:1B90=E1 29 CD D8 28 C8 CD 9E :0A
:1B98=2A 38 0C CD 35 29 28 13 :D4
:1BA0=3C 32 0F 00 C3 E1 29 2B :75
:1BA8=7E FE 0D C0 23 22 B4 14 :56
:1BB0=C3 EC 27 CD 04 28 CD EC :88
:1BB8=27 CD 43 1C CD 0D 28 C3 :18
:1BC0=E1 29 2A B4 14 CD 22 1C :07
:1BC8=38 0E 2B CD E1 28 DA F2 :13
:1BD0=1B CD 22 1C 28 16 30 F2 :86
:1BD8=2B CD E1 28 DA F2 1B CD :B5
:1BE0=22 1C 28 08 38 F2 22 B4 :6E
:1BE8=14 C3 1C 2A 22 B4 14 C3 :CA
:1BF0=55 1B 22 B4 14 C3 16 28 :5B
:1BF8=CD D8 28 C8 FE 0D 20 06 :C6
:1C00=CD CA 2A C3 32 1B CD 0F :AD
:1C08=1C 22 B4 14 C3 1C 2A 2A :39
:1C10=B4 14 CD 22 1C C8 23 30 :EE
:1C18=F9 CD 22 1C C8 23 38 F9 :20
:1C20=2B C9 7E B7 C8 FE 0D C8 :C4
:1C28=FE 30 38 13 FE 3A 38 0C :F5
:1C30=FE 80 30 08 E6 1F 28 07 :EA
:1C38=FE 1B 30 03 F6 FF C9 F6 :00
:1C40=FF 37 C9 F5 E5 3A 1E 00 :31
:1C48=32 0E 00 CD 30 28 E1 F1 :37
:1C50=C9 CD 8A 29 D8 3E 01 32 :92
:1C58=46 FB CD 04 28 CD 16 28 :45
:1C60=CD 94 29 CD 6A 1B 3A 39 :4F
:1C68=FB 47 3A 17 00 B8 CA 79 :8E
:1C70=1C 04 78 32 39 FB C3 0D :CE
:1C78=28 CD 72 1C CD 72 2A D8 :C4

```

```

:1C80=EB CD 75 2A C3 E1 29 CD :F1
:1C88=D8 28 C8 CD A4 29 38 25 :BF
:1C90=CD 04 28 3A 17 00 32 0F :8B
:1C98=00 CD EC 27 CD 43 1C 3A :46
:1CA0=39 FB 47 3A 16 00 B8 28 :AB
:1CA8=03 05 18 C6 CD 72 1C CD :0E
:1CB0=9E 2A C3 E1 29 2B 7E FE :3C
:1CB8=0D C0 18 D4 CD 8A 29 D8 :11
:1CC0=22 B4 14 CD 25 28 32 46 :7C
:1CC8=FB CD 94 29 22 B4 14 E5 :54
:1CD0=CD 04 28 CD 16 28 CD 73 :44
:1CD8=28 CD 0D 28 E1 C3 BA 29 :B1
:1CE0=CD 8A 29 CD 25 28 32 46 :12
:1CE8=FB CD 67 29 D8 18 DD DD :02
:1CF0=CB 00 4E C0 2A B4 14 DD :A8
:1CF8=CB 00 46 20 09 7E FE 09 :BF
:1D00=28 15 FE 0E 30 11 E5 CD :3C
:1D08=06 1A E1 CD 28 29 DA FE :F7
:1D10=19 CD 30 28 C3 1C 2A CD :14
:1D18=28 29 DA FE 19 3E 09 CD :56
:1D20=13 00 CD 72 2A C3 E1 29 :49
:1D28=CD D8 28 C8 F5 CD 3F 27 :BD
:1D30=F1 2A B4 14 CD 04 28 FE :DA
:1D38=0D 28 06 CD 30 28 C3 0D :30
:1D40=28 CD 73 28 CD 0D 28 C9 :5B
:1D48=AF 32 3F FB CD 53 2A 22 :87
:1D50=B4 14 E5 CD 47 27 CD 04 :B9
:1D58=28 3A 1E 00 32 0E 00 CD :8D
:1D60=73 28 CD 0D 28 E1 C3 E1 :22
:1D68=29 AF 32 3F FB CD CF 28 :08
:1D70=CA 28 1D FE 00 C8 CD 0F :B1
:1D78=1C ED 5B B4 14 CD 5C 2A :7F
:1D80=CD 47 27 CD 04 28 CD 30 :31
:1D88=28 C3 0D 28 AF 32 3F FB :3B
:1D90=CD D8 28 C8 CD 64 2A CD :BD
:1D98=47 27 C3 FC 27 AF 32 3F :74
:1DA0=FB 0E 0D CD 85 27 DD CB :37
:1DA8=00 5E C0 CD 04 28 2A B4 :F5
:1DB0=14 C3 41 1D DD CB 00 4E :2B
:1DB8=C0 CD 06 1A DD CB 00 46 :9B
:1DC0=28 1D CD D8 28 CA EC 27 :EF
:1DC8=CD FC 27 CD EC 27 2A B4 :AE
:1DD0=14 CD 04 28 3E 0F CD 13 :3A
:1DD8=00 CD 30 28 C3 0D 28 CD :EA
:1DE0=9E 2A 22 B4 14 2B 7E FE :59
:1DE8=0D 20 08 CD 35 29 28 DB :63
:1DF0=C3 EC 27 0E 0D CD C4 27 :A9
:1DF8=CD EC 27 C3 FC 27 DD CB :6E
:1E00=00 46 28 05 DD CB 00 86 :A1
:1E08=C9 DD CB 00 C6 C9 FD 21 :1E
:1E10=17 1E 06 09 C3 E5 23 52 :61
:1E18=32 1E 43 44 1E 46 A2 1E :FB
:1E20=41 19 21 42 56 1E 4B 5C :D8
:1E28=1E 4C 04 23 59 8C 1D 4D :E0
:1E30=78 1A CD A4 29 2A B0 14 :1A
:1E38=22 B4 14 CD FD 2D D2 16 :C9
:1E40=28 C3 E5 2D CD A4 29 2A :C1
:1E48=B6 14 22 B4 14 CD FD 2D :AB
:1E50=D2 02 2A C3 EB 2D FD 21 :F7
:1E58=67 2E 18 04 FD 21 6A 2E :67
:1E60=CD A4 29 2A B4 14 7E FD :07
:1E68=BE FF C8 CD B6 24 38 06 :6A
:1E70=22 B4 14 C3 BC 2D 11 7C :23
:1E78=1E C3 68 2D 20 2A 2A 20 :0A
:1E80=4D 61 72 6B 65 72 20 4E :D0
:1E88=6F 74 20 46 6F 75 6E 64 :FF
:1E90=20 2A 2A 20 50 72 65 73 :2E
:1E98=73 20 52 65 74 75 72 6E :13
:1EA0=1A 00 CD 0F 2D DD CB 01 :CC
:1EA8=AE CD 4B 1F DA 91 2D B7 :34

```

つづく



## リスト4-13 つづき

```

:1EB0=CA 91 2D 32 B6 FA CD 50 :87
:1EB8=20 DA 91 2D DD CB 01 FE :5F
:1EC0=3A B4 FA 32 B5 FA 2A B4 :A7
:1EC8=14 CD DD 1E DA 70 2D DD :30
:1ED0=35 03 20 F5 22 B4 14 CD :04
:1ED8=B5 28 C3 BC 2D DD CB 01 :32
:1EE0=66 20 2A EB CD BE 2A EB :3B
:1EE8=11 B9 FA 1A 13 ED B1 28 :B7
:1EF0=02 37 C9 1A 13 B7 C8 BE :6C
:1EF8=23 0B F5 78 B1 28 07 F1 :6C
:1F00=28 F1 2B 03 18 E2 F1 CA :FC
:1F08=3C 1C C3 3F 1C 2B CD E1 :4F
:1F10=28 E5 21 B9 FA 3A B6 FA :CB
:1F18=5F 16 00 19 2B 22 25 1F :1F
:1F20=E1 CD B4 2A 11 00 00 1A :B7
:1F28=1B ED B9 28 02 37 C9 1A :05
:1F30=1B B7 28 15 BE 2B 0B F5 :F8
:1F38=78 B1 28 07 F1 28 F0 23 :84
:1F40=03 18 E1 F1 C2 3F 1C F6 :00
:1F48=FF 23 C9 21 00 01 11 99 :B7
:1F50=1F CD 7E 2B CD 0B 00 21 :8E
:1F58=B9 FA AF 32 B8 FA 06 31 :7D
:1F60=0E 00 CD A3 38 FE 03 CA :81
:1F68=3F 1C FE 08 28 1C FE 09 :AC
:1F70=28 08 FE 0D 28 0E FE 20 :8F
:1F78=38 E8 CD C8 04 77 23 0C :5F
:1F80=78 B9 20 DE 36 00 F6 FF :5A
:1F88=79 C9 CD 8F 1F 18 D3 79 :21
:1F90=B7 C8 0D 2B 3E 08 C3 13 :D3
:1F98=00 46 69 6E 64 3F 20 00 :E0
:1FA0=0D 52 65 70 6C 61 63 65 :C9
:1FA8=20 77 69 74 68 3F 20 00 :3B
:1FB0=0D 4F 70 74 69 6F 6E 73 :F9
:1FB8=3F 20 28 3F 20 46 6F 72 :0D
:1FC0=20 49 6E 66 6F 29 20 00 :F5
:1FC8=20 20 20 4E 6F 72 6D 61 :5D
:1FD0=6C 6C 79 20 50 72 65 73 :0B
:1FD8=73 20 52 65 74 75 72 6E :13
:1FE0=20 6F 6E 6C 79 2C 6F 72 :EF
:1FE8=20 65 6E 74 65 72 20 6F :CD
:1FF0=6E 65 20 6F 72 20 6D 6F :D0
:1FF8=72 65 20 6F 66 3A 0D 6E :81
:2000=75 6D 62 65 72 3D 72 65 :2F
:2008=70 65 61 74 20 63 6F 75 :11
:2010=6E 74 2C 47 3D 72 65 70 :D9
:2018=6C 61 63 65 20 69 6E 20 :AC
:2020=65 6E 74 69 72 65 20 66 :0D
:2028=69 6C 65 2C 4E 3D 72 65 :C8
:2030=70 6C 61 63 65 20 61 73 :F9
:2038=6B 69 6E 67 2C 42 3D 73 :C7
:2040=65 61 72 63 68 20 42 61 :C6
:2048=63 6B 77 61 72 64 73 00 :EF
:2050=11 B0 1F CD 0B 00 2A 0E :F0
:2058=00 22 30 FB 3E 01 32 B4 :72
:2060=FA DD CB 01 96 DD CB 01 :E2
:2068=9E DD CB 01 A6 21 1E FB :27
:2070=0E 00 18 03 CD 8F 1F CD :71
:2078=A3 38 FE 03 CA 3F 1C FE :FF
:2080=08 28 F1 FE 0D 28 0F FE :61
:2088=20 38 EC CD 13 00 77 0C :A7
:2090=23 3E 09 B9 20 E1 36 00 :5A
:2098=21 1E FB 7E 23 CD 51 14 :0D
:20A0=B7 C8 FE 30 38 F5 FE 3A :12
:20A8=38 3C FE 42 28 32 FE 47 :53
:20B0=28 22 FE 4E 28 24 FE 3F :1F
:20B8=20 E1 21 00 04 11 C8 1F :1E
:20C0=CD 7E 2B CD 0B 00 CD FC :17
:20C8=27 2A 30 FB 22 0E 00 CD :79
:20D0=FC 27 18 88 DD CB 01 D6 :42
:20D8=18 C1 DD CB 01 DE 18 BB :33

```

```

:20E0=DD CB 01 E6 18 B5 D6 30 :62
:20E8=32 B4 FA 47 7E 23 FE 30 :F6
:20F0=38 AB FE 3A 30 A7 D6 30 :F8
:20F8=4F 78 87 87 80 87 81 32 :8F
:2100=B4 FA 18 97 DD CB 01 7E :84
:2108=C8 CD 04 28 CD A4 29 DD :38
:2110=CB 01 6E CA BC 1E C3 48 :E9
:2118=21 CD 0F 2D DD CB 01 EE :C1
:2120=CD 4B 1F DA 91 2D B7 CA :50
:2128=91 2D 32 B6 FA 11 A0 1F :70
:2130=CD 0B 00 21 EC FA CD 5E :0A
:2138=1F DA 91 2D 32 B7 FA CD :67
:2140=50 20 DA 91 2D CD F5 2C :F6
:2148=DD CB 01 B6 3A B4 FA 32 :79
:2150=B5 FA DD CB 01 FE DD CB :FE
:2158=01 56 20 2A CD A4 29 CD :08
:2160=E9 21 38 61 CD 12 22 38 :DC
:2168=10 DD CB 00 5E 20 0A CD :0D
:2170=4A 2E 38 05 DD 35 03 20 :EA
:2178=E3 CD 5A 39 CD B5 28 CD :BA
:2180=F4 2D CD 70 2B C9 DD CB :FA
:2188=01 A6 CD A4 29 2A B4 14 :33
:2190=22 32 FB 2A B0 14 22 B4 :13
:2198=14 2A 3D FB 22 34 FB 21 :E8
:21A0=01 00 22 3D FB 2A 49 FB :C9
:21A8=22 36 FB CD E9 21 38 1E :80
:21B0=CD 12 22 38 C4 DD CB 00 :A5
:21B8=5E 20 BE CD 4A 2E 38 B9 :72
:21C0=CD A4 29 18 E6 DD CB 01 :41
:21C8=76 CA 70 2D 18 AB DD CB :48
:21D0=01 76 20 A5 2A 32 FB 22 :B5
:21D8=B4 14 2A 34 FB 22 3D FB :7B
:21E0=2A 36 FB 22 49 FB C3 70 :F4
:21E8=2D CD 70 2B 2A B4 14 CD :54
:21F0=DD 1E D8 DD CB 01 F6 DD :4F
:21F8=CB 01 66 28 07 3A B6 FA :4B
:2200=5F 16 00 19 22 B4 14 E5 :5D
:2208=CD 21 2B CD D1 2B E1 C3 :86
:2210=BC 2D DD CB 01 5E 20 5C :6C
:2218=CD 04 28 21 3F 00 11 F5 :5F
:2220=22 CD 79 2B CD 0B 00 3E :A9
:2228=17 32 26 00 CD BA 04 3E :38
:2230=07 32 26 00 2A 0E 00 22 :B9
:2238=30 FB CD 0D 28 CD A3 38 :D5
:2240=CD 51 14 FE 4E 28 1C FE :C0
:2248=59 28 09 FE 03 20 EE CD :66
:2250=63 22 37 C9 2A 30 FB 22 :FC
:2258=0E 00 CD 13 00 CD 0D 28 :F0
:2260=CD 87 22 CD 04 28 21 3F :CF
:2268=00 CD 79 2B CD FC 27 CD :2E
:2270=0D 28 B7 C9 CD 87 22 B7 :E2
:2278=C9 2A B4 14 3A B7 FA 4F :F5
:2280=06 00 B7 ED 42 EB C9 3A :DA
:2288=B6 FA B7 37 C8 47 3A B7 :9E
:2290=FA B7 CA E0 22 4F 90 28 :84
:2298=23 30 07 78 91 CD E2 22 :34
:22A0=18 1A 4F 06 00 ED 43 42 :F9
:22A8=FB CD A0 27 DD CB 00 5E :95
:22B0=C0 ED 4B 42 FB 2A B4 14 :27
:22B8=09 22 B4 14 CD 79 22 21 :7C
:22C0=EC FA D5 ED B0 E1 22 B4 :0F
:22C8=14 D5 E5 CD 1C 2A E1 D1 :93
:22D0=CD 30 28 DD CB 01 66 20 :54
:22D8=04 ED 53 B4 14 C3 1C 2A :15
:22E0=48 3E 4F 06 00 ED 43 42 :4D
:22E8=FB 2A B4 14 B7 ED 42 22 :F5
:22F0=B4 14 C3 47 27 52 65 70 :20
:22F8=6C 61 63 65 20 28 59 2F :65
:2300=4E 29 3F 00 CD 0F 2D 21 :E0
:2308=02 02 CD 7E 2B 11 25 23 :D3

```

つづく



リスト4-13

つづき

:2310=CD 0B 00 CD 34 23 DA 85 :5B  
 :2318=2D CD 96 23 22 B4 14 CD :6A  
 :2320=B5 28 C3 BC 2D 4C 69 6E :AC  
 :2328=65 20 4E 75 6D 62 65 72 :EE  
 :2330=20 3F 3A 00 21 00 00 22 :DC  
 :2338=40 FB 21 1E FB 0E 00 18 :9B  
 :2340=03 CD 8F 1F CD A3 38 FE :24  
 :2348=03 CA 3F 1C FE 08 28 F1 :47  
 :2350=FE 0D 28 13 FE 30 38 EC :98  
 :2358=FE 3A 30 E8 CD 13 00 77 :A7  
 :2360=23 0C 3E 05 B9 20 DD 36 :5E  
 :2368=00 79 B7 28 0F 11 1E FB :91  
 :2370=1A 13 B7 C8 D6 30 CD 85 :04  
 :2378=23 30 F5 C9 21 01 00 22 :55  
 :2380=40 FB F6 FF C9 2A 40 FB :5E  
 :2388=4D 44 29 29 09 29 4F 06 :6A  
 :2390=00 09 22 40 FB C9 ED 5B :77  
 :2398=3D FB 2A 40 FB B7 ED 52 :93  
 :23A0=20 04 2A B4 14 C9 38 11 :28  
 :23A8=22 40 FB CD BA 2A EB CD :C6  
 :23B0=A2 2A D8 CD D3 23 20 F6 :7D  
 :23B8=C9 2A 40 FB EB B7 ED 52 :0F  
 :23C0=23 22 40 FB ED 5B B4 14 :90  
 :23C8=EB CD 75 2A D8 CD D3 23 :F2  
 :23D0=20 F6 C9 E5 2A 40 FB 2B :54  
 :23D8=7C B5 22 40 FB E1 C9 FD :35  
 :23E0=21 11 24 06 0B CD CF 2C :2F  
 :23E8=3E 5E CD 13 00 3E 40 81 :7B  
 :23F0=CD 13 00 21 C0 2C E5 CD :9F  
 :23F8=B0 2C CD 0D 28 D9 FD E5 :99  
 :2400=E1 BE 23 28 06 23 23 10 :46  
 :2408=F8 D9 C9 5E 23 56 D5 D9 :1F  
 :2410=C9 42 32 24 4B 38 24 59 :61  
 :2418=79 25 43 D1 25 48 01 25 :45  
 :2420=56 11 26 53 6C 2E 51 B1 :7C  
 :2428=26 57 75 2E 52 7E 2E 50 :6E  
 :2430=ED 26 FD 21 67 2E 18 04 :E2  
 :2438=FD 21 6A 2E CD A4 29 CD :1D  
 :2440=B6 24 DA 9B 24 ED 5B B4 :6F  
 :2448=14 CD 44 2E C8 D2 A8 24 :B9  
 :2450=CD D7 24 C0 CD D0 24 CD :16  
 :2458=E2 24 2B 22 B4 14 EB CD :D3  
 :2460=FD 2D DA 97 24 CD 1E 2E :D8  
 :2468=CD 2F 2E D2 8E 24 E5 D5 :68  
 :2470=CD 97 24 D1 E1 ED 53 32 :AC  
 :2478=FB 22 B4 14 E5 CD 02 2A :C3  
 :2480=E1 CD 30 28 CD 0D 28 2A :32  
 :2488=32 FB 22 B4 14 C9 2A 28 :32  
 :2490=FB CD 43 1C C3 02 2A EB :01  
 :2498=C3 F8 24 CD D7 24 C0 2A :91  
 :24A0=B4 14 CD C9 24 C3 F8 24 :61  
 :24A8=CD D7 24 C0 CD D0 24 23 :6C  
 :24B0=CD E2 24 C3 5B 24 ED 5B :5D  
 :24B8=B0 14 CD BE 2A EB FD 7E :DF  
 :24C0=FF ED B1 28 02 37 C9 2B :F2  
 :24C8=B7 FD 75 00 FD 74 01 C9 :64  
 :24D0=FD 6E 00 FD 66 01 C9 FD :95  
 :24D8=4E FF CD 85 27 DD CB 00 :6E  
 :24E0=5E C9 ED 5B B4 14 ED 53 :77  
 :24E8=32 FB 22 B4 14 CD 3F 27 :4A  
 :24F0=2A 32 FB ED 5B B4 14 C9 :30  
 :24F8=CD 04 28 CD 30 28 C3 0D :EE  
 :2500=28 CD A4 29 FD 21 67 2E :75  
 :2508=CD 0F 25 FD 21 6A 2E 2A :E1  
 :2510=B4 14 7E FD BE FF CA 51 :1B  
 :2518=25 22 32 FB CD B6 24 D8 :F3  
 :2520=22 B4 14 CD 3F 27 ED 5B :65  
 :2528=32 FB CD 44 2E C8 30 01 :65  
 :2530=1B ED 53 B4 14 CD FD 2D :1A  
 :2538=D8 ED 53 32 FB 22 B4 14 :2F

:2540=E5 CD 02 2A E1 CD 30 28 :E4  
 :2548=2A 32 FB 22 B4 14 C3 02 :06  
 :2550=2A CD 3F 27 CD 30 28 C3 :45  
 :2558=02 2A FD 21 67 2E CD B6 :62  
 :2560=24 D8 FD 21 6A 2E CD B6 :35  
 :2568=24 D8 ED 5B 67 2E 13 CD :B9  
 :2570=44 2E CA 3F 1C 1B C3 44 :B9  
 :2578=2E 3E 02 32 3F FB CD 5A :01  
 :2580=25 DA 65 2D CD C1 2A ED :36  
 :2588=43 42 FB CD A4 29 2A B4 :F8  
 :2590=14 CD 94 26 FE 01 20 06 :C0  
 :2598=CD C8 25 C3 F1 2D B7 28 :7A  
 :25A0=19 CD C2 25 ED 5B 42 FB :52  
 :25A8=2A 32 FB B7 ED 52 22 B4 :23  
 :25B0=14 CD 6D 26 C2 F1 2D C3 :17  
 :25B8=02 2A CD C2 25 2A 32 FB :37  
 :25C0=18 EC 2A B4 14 22 32 FB :45  
 :25C8=2A 67 2E 22 B4 14 C3 47 :B3  
 :25D0=27 CD 5A 25 DA 65 2D E5 :C4  
 :25D8=2A B4 14 CD 94 26 E1 FE :58  
 :25E0=01 C8 CD C1 2A 0B 0B ED :84  
 :25E8=43 42 FB CD A0 27 DD CB :BC  
 :25F0=00 5E C0 EB 23 3A 2C FB :8D  
 :25F8=FE 02 28 04 FE 01 C8 09 :FC  
 :2600=ED 5B B4 14 D5 ED B0 E1 :63  
 :2608=CD 04 28 CD 73 28 C3 0D :31  
 :2610=28 3E 02 32 3F FB CD 5A :FB  
 :2618=25 DA 65 2D E5 2A B4 14 :68  
 :2620=CD 94 26 E1 FE 01 C8 CD :FC  
 :2628=C1 2A ED 43 42 FB CD A0 :C5  
 :2630=27 DD CB 00 5E C0 EB 3A :12  
 :2638=2C FB FE 01 C8 FE 02 28 :16  
 :2640=01 09 E5 ED 5B B4 14 ED :EC  
 :2648=53 32 FB ED B0 E1 22 B4 :D4  
 :2650=14 CD 47 27 2A 32 FB 3A :E0  
 :2658=2C FB FE 01 C8 B7 28 07 :D4  
 :2660=ED 4B 42 FB B7 ED 42 22 :7D  
 :2668=B4 14 C3 F1 2D 2A 6A 2E :6B  
 :2670=CD FD 2D 30 19 2A 67 2E :FF  
 :2678=CD FD 2D 30 11 ED 5B 4D :CD  
 :2680=FB CD 44 2E 30 0C 2A 6A :0A  
 :2688=2E CD 44 2E 38 04 3E 01 :E8  
 :2690=B7 C9 AF C9 D5 ED 5B 67 :7C  
 :2698=2E AF CD 44 2E 38 0D 3C :9D  
 :26A0=ED 5B 6A 2E CD 44 2E 28 :47  
 :26A8=03 38 01 3C 32 2C FB D1 :A2  
 :26B0=C9 CD 0F 2D 21 02 02 11 :08  
 :26B8=D4 26 CD 7E 2B CD 0B 00 :48  
 :26C0=CD A3 38 CD 13 00 CD 51 :A6  
 :26C8=14 D6 59 C2 95 2D CD A4 :38  
 :26D0=28 C3 7F 19 0D 41 62 61 :94  
 :26D8=6E 64 6F 6E 20 61 20 46 :96  
 :26E0=69 6C 65 20 3F 20 28 59 :3A  
 :26E8=2F 4E 29 20 00 CD 5A 25 :12  
 :26F0=DA 65 2D 13 2B 18 07 ED :B6  
 :26F8=5B B0 14 2A B6 14 CD C1 :A1  
 :2700=2A EB CD 39 27 3E 0D CD :5A  
 :2708=23 27 7E B7 28 0A CD 23 :A1  
 :2710=27 D8 23 0B 78 B1 20 F2 :68  
 :2718=3A 2F FB B7 C8 3E 0C CD :FA  
 :2720=6A 39 C9 FE 0D C2 6A 39 :DC  
 :2728=CD 6A 39 D8 3A 2F FB 3C :E8  
 :2730=FE 3C 20 06 3E 0C CD 6A :E1  
 :2738=39 AF 32 2F FB B7 C9 E5 :A9  
 :2740=21 01 00 22 42 FB E1 C5 :27  
 :2748=D5 E5 F5 2A B4 14 E5 ED :73  
 :2750=5B 42 FB 19 CD F5 28 EB :86  
 :2758=CD BE 2A E1 EB ED B0 1B :39  
 :2760=ED 53 B6 14 DD CB 00 8E :40  
 :2768=C3 D0 27 C5 D5 E5 F5 21 :4F  
 :2770=01 00 22 42 FB CD A0 27 :F4

つづく



## リスト4-13 つづき

```

:2778=DD CB 00 5E C2 D0 27 CD :8C
:2780=D5 27 C3 D0 27 C5 D5 E5 :35
:2788=F5 21 01 00 22 42 FB CD :43
:2790=A0 27 DD CB 00 5E C2 D0 :5F
:2798=27 2A B4 14 71 C3 D0 27 :44
:27A0=C5 D5 E5 F5 CD BA 2A E5 :0A
:27A8=ED 5B 42 FB CD F9 2A 38 :AD
:27B0=0B 19 22 B6 14 EB E1 ED :C9
:27B8=B8 C3 D0 27 E1 DD CB 00 :FB
:27C0=DE C3 D0 27 C5 D5 E5 F5 :0C
:27C8=CD D5 27 22 B6 14 36 00 :EB
:27D0=F1 E1 D1 C1 C9 2A B4 14 :1F
:27D8=71 23 CD F5 28 22 B4 14 :68
:27E0=D0 DD CB 00 DE C9 3E 07 :64
:27E8=32 26 00 C9 F5 3E 0D CD :2E
:27F0=13 00 F1 C9 F5 3E 0C CD :D9
:27F8=13 00 F1 C9 F5 3E 05 CD :D2
:2800=13 00 F1 C9 E5 2A 0E 00 :EA
:2808=22 38 FB E1 C9 E5 2A 38 :46
:2810=FB 22 0E 00 E1 C9 F5 3A :04
:2818=1E 00 32 0E 00 3A 16 00 :AE
:2820=32 0F 00 F1 C9 3A 16 00 :4B
:2828=47 3A 17 00 90 CB 2F C9 :EB
:2830=CD FC 27 CD 1D 29 28 2B :56
:2838=7E B7 28 25 FE 0D 28 1F :D4
:2840=FE 1C 28 15 FE 1D 28 11 :AB
:2848=FE 09 20 07 CD 28 29 38 :84
:2850=12 3E 09 CD 13 00 23 18 :74
:2858=DA CD C8 04 23 18 D4 B7 :39
:2860=C9 37 C9 F5 3A 1F 00 3D :54
:2868=32 0E 00 3E 1C CD C8 04 :33
:2870=F1 B7 C9 C5 D5 E5 3A 0F :39
:2878=00 47 3A 17 00 3C 90 47 :AB
:2880=18 03 CD 13 00 CD 30 28 :20
:2888=38 0B 7E 23 B7 28 06 FE :C7
:2890=0D 20 F7 10 ED CD 9C 28 :B2
:2898=E1 D1 C1 C9 F5 3E 1A CD :56
:28A0=13 00 F1 C9 E5 F5 AF 32 :88
:28A8=B2 FA 2A B0 14 36 00 22 :F2
:28B0=B6 14 F1 E1 C9 F5 3E 00 :98
:28B8=32 1E 00 3A B8 14 3C 32 :C4
:28C0=16 00 3E 4F 32 1F 00 3A :2E
:28C8=B9 14 32 17 00 F1 C9 E5 :B5
:28D0=2A B4 14 7E FE 0D E1 C9 :25
:28D8=E5 2A B4 14 7E FE 00 E1 :34
:28E0=C9 D5 E5 ED 5B B0 14 B7 :46
:28E8=ED 52 E1 38 04 28 02 D1 :57
:28F0=C9 EB D1 37 C9 D5 ED 5B :A2
:28F8=B2 14 E5 B7 ED 52 E1 28 :AA
:2900=02 30 EE B7 D1 C9 C5 47 :7D
:2908=3A 1E 00 B8 28 02 30 06 :70
:2910=3A 1F 00 B8 30 03 37 C1 :3C
:2918=C9 78 B7 C1 C9 C5 3A 1F :A0
:2920=00 47 3A 0E 00 B8 C1 C9 :D1
:2928=C5 3A 0E 00 47 3A 1F 00 :AD
:2930=D6 08 B8 C1 C9 C5 3A 17 :36
:2938=00 47 3A 0F 00 B8 C1 C9 :D2
:2940=C5 3A 16 00 47 3A 0F 00 :A5
:2948=B8 C1 C9 7E B7 28 15 FE :B2
:2950=09 28 07 FE 0D 28 0D 1C :94
:2958=18 06 7B E6 F8 C6 08 5F :A4
:2960=7B 23 B7 C9 7B 37 C9 EB :84
:2968=CD BE 2A EB CD A2 2A D8 :11
:2970=CD 82 29 20 F6 C9 3A 16 :A7
:2978=00 47 3A 0F 00 90 32 46 :98
:2980=FB C9 3A 46 FB 3D 32 46 :F4
:2988=FB C9 CD 72 2A 38 11 CD :43
:2990=76 29 28 0C EB CD 75 2A :2A
:2998=38 06 CD 82 29 20 F5 B7 :82
:29A0=22 4D FB C9 CD 8A 29 3A :ED
:29A8=16 00 47 3A 17 00 90 3C :7A
:29B0=32 46 FB CD 67 29 22 4F :41
:29B8=FB C9 CD 76 29 28 22 CD :47
:29C0=67 29 30 1D 22 B4 14 3A :01
:29C8=46 FB 4F 3A 0F 00 91 80 :EA
:29D0=32 0F 00 2B 7E FE 0D C2 :B7
:29D8=1C 2A 3A 1E 00 32 0E 00 :DE
:29E0=C9 3A 0E 00 4F 3A 1E 00 :B8
:29E8=B9 28 0A 1E 00 CD 4B 29 :4A
:29F0=38 03 B9 38 F8 CD 06 29 :20
:29F8=32 0E 00 22 B4 14 C9 1C :0F
:2A00=18 0F ED 5B 4D FB 2A B4 :95
:2A08=14 CD C1 2A EB 3A 16 00 :07
:2A10=5F 3E 0D ED B1 EA FF 29 :5A
:2A18=7B 32 0F 00 CD 72 2A ED :12
:2A20=5B B4 14 EB E5 B7 ED 52 :E9
:2A28=44 4D E1 EB 28 14 1E 00 :B7
:2A30=CD 4B 29 32 0E 00 CD 1D :6B
:2A38=29 30 0E 0B 78 B1 20 F0 :AB
:2A40=2B C9 3A 1E 00 32 0E 00 :8C
:2A48=C9 3A 1F 00 32 0E 00 22 :84
:2A50=B4 14 C9 CD 72 2A EB D5 :BA
:2A58=CD 9E 2A D1 B7 ED 52 22 :7E
:2A60=42 FB EB C9 ED 5B B4 14 :01
:2A68=D5 CD 9E 2A D1 38 ED 2B :8B
:2A70=18 EA 2A B4 14 CD E1 28 :CA
:2A78=D8 7E FE 0D 20 06 2B 7E :30
:2A80=FE 0D 28 14 ED 5B B0 14 :53
:2A88=CD C1 2A 3E 0D ED B9 28 :D1
:2A90=05 2A B0 14 37 C9 B7 23 :CD
:2A98=54 5D 23 C3 E1 28 CD BA :27
:2AA0=2A EB 3E 0D ED B1 28 05 :2B
:2AA8=2A B6 14 37 C9 B7 54 5D :5C
:2AB0=C9 2A B4 14 ED 5B B0 14 :C7
:2AB8=18 07 ED 5B B4 14 2A B6 :0F
:2AC0=14 E5 B7 ED 52 23 44 4D :A3
:2AC8=E1 C9 E5 2A B4 14 23 CD :71
:2AD0=F5 28 22 B4 14 E1 C9 E5 :96
:2AD8=2A B4 14 2B CD E1 28 22 :15
:2AE0=B4 14 E1 C9 ED 5B B6 14 :84
:2AE8=2A B2 14 B7 ED 52 22 44 :4C
:2AF0=FB 7D B4 C0 DD CB 00 CE :62
:2AF8=C9 E5 D5 C5 CD E4 2A ED :10
:2B00=5B 42 FB B7 ED 52 C1 D1 :20
:2B08=E1 C9 3A 3F FB B7 CA A4 :43
:2B10=19 FE 02 28 06 CD 21 2B :60
:2B18=C3 A4 19 CD 5A 2B C3 A4 :39
:2B20=19 2A B4 14 ED 5B 49 FB :97
:2B28=CD 44 2E C8 38 28 3E 01 :A6
:2B30=F5 CD C1 2A EB 11 00 00 :A9
:2B38=CD 4B 2B F1 2A 3D FB B7 :4D
:2B40=20 04 ED 52 18 01 19 22 :B7
:2B48=3D FB C9 3E 0D 18 01 13 :78
:2B50=ED B1 EA 4F 2B C9 AF EB :65
:2B58=18 D6 2A B4 14 ED 5B B0 :D8
:2B60=14 CD C1 2A EB 11 01 00 :C9
:2B68=CD 4B 2B ED 53 3D FB C9 :84
:2B70=E5 2A B4 14 22 49 FB E1 :1E
:2B78=C9 3A B8 14 18 03 3A 16 :3A
:2B80=00 84 67 3A 1E 00 85 6F :37
:2B88=22 0E 00 C9 3E 01 32 3C :A6
:2B90=FB 01 10 27 CD AF 2B 01 :DB
:2B98=E8 03 CD AF 2B 01 64 00 :F7
:2BA0=CD AF 2B 01 0A 00 CD AF :2E
:2BA8=2B 7D F6 30 12 13 C9 3E :FA
:2BB0=30 B7 ED 42 38 03 3C 18 :A5
:2BB8=F8 09 FE 30 28 07 12 13 :83
:2BC0=AF 32 3C FB C9 F5 3A 3C :4C
:2BC8=FB B7 28 02 F1 C9 F1 18 :9F

```

つづく



リスト4-13 つづき

:2BD0=ED C5 D5 E5 F5 CD E4 2A :3C  
 :2BD8=2A 0E 00 22 3A FB 21 00 :B0  
 :2BE0=FF 11 01 FF 3A 1E 00 4F :B7  
 :2BE8=3A 1F 00 91 4F 0C 06 00 :4B  
 :2BF0=ED B0 21 00 FF 11 7B 2C :75  
 :2BF8=CD 73 2C EB 3A 3A FB 3C :02  
 :2C00=6F 26 00 CD 8C 2B EB 11 :15  
 :2C08=85 2C CD 73 2C EB 2A 3D :6F  
 :2C10=FB CD 8C 2B EB 11 8C 2C :33  
 :2C18=CD 73 2C EB D5 CD B1 2A :D4  
 :2C20=C5 E1 D1 CD 8C 2B EB 36 :1C  
 :2C28=20 23 DD CB 00 46 20 05 :56  
 :2C30=11 99 2C 18 03 11 91 2C :BF  
 :2C38=CD 73 2C DD CB 00 4E 28 :8A  
 :2C40=06 11 A4 2C CD 73 2C 36 :89  
 :2C48=05 23 36 00 21 00 00 CD :4C  
 :2C50=79 2B 11 00 FF CD 0B 00 :8C  
 :2C58=2A 3A FB 22 0E 00 DD CB :37  
 :2C60=00 5E 28 0A CD F7 07 CD :28  
 :2C68=F7 07 DD CB 00 9E F1 E1 :16  
 :2C70=D1 C1 C9 1A B7 C8 13 77 :7E  
 :2C78=23 18 F8 20 20 20 20 20 :D3  
 :2C80=43 4F 4C 3D 00 20 4C 49 :D0  
 :2C88=4E 45 3D 00 20 46 43 3D :B6  
 :2C90=00 49 4E 53 45 52 54 20 :F5  
 :2C98=00 4F 56 45 52 57 52 49 :2E  
 :2CA0=54 45 20 00 42 55 46 46 :DC  
 :2CA8=45 52 20 46 55 4C 4C 00 :EA  
 :2CB0=CD A3 38 FE 20 30 02 C6 :BE  
 :2CB8=40 CD 13 00 CD 51 14 C9 :1B  
 :2CC0=CD CF 2C CD BA 04 CD BA :DA  
 :2CC8=04 CD BA 04 C3 0D 28 CD :54  
 :2CD0=04 28 3A B8 14 67 2E 00 :C7  
 :2CD8=22 0E 00 C9 F5 3E 00 32 :5E  
 :2CE0=1E 00 3A B8 14 3C 32 16 :A8  
 :2CE8=00 C6 06 32 17 00 3E 4F :A2  
 :2CF0=32 1F 00 F1 C9 F5 AF 32 :E1  
 :2CF8=1E 00 3A B8 14 C6 06 32 :22  
 :2D00=16 00 3E 4F 32 1F 00 3A :2E  
 :2D08=B9 14 32 17 00 F1 C9 CD :9D  
 :2D10=A4 29 CD DC 2C C3 F4 27 :80  
 :2D18=0D 20 20 2A 2A 20 4E 6F :7E  
 :2D20=74 20 46 6F 75 6E 64 20 :B0  
 :2D28=2A 2A 20 22 00 22 20 50 :28  
 :2D30=72 65 73 73 20 52 65 74 :08  
 :2D38=75 72 6E 1A 00 20 20 2A :D9  
 :2D40=2A 20 4D 61 72 6B 65 72 :AC  
 :2D48=20 53 65 74 20 45 72 72 :95  
 :2D50=6F 72 20 2A 2A 20 2C 50 :F1  
 :2D58=72 65 73 73 20 52 65 74 :08  
 :2D60=75 72 6E 1A 00 11 3D 2D :EA  
 :2D68=CD 04 28 CD EC 27 18 12 :03  
 :2D70=CD EC 27 11 18 2D CD 0B :0E  
 :2D78=00 11 B9 FA CD 0B 00 11 :AD  
 :2D80=2D 2D CD 0B 00 CD F7 07 :FD  
 :2D88=CD A3 38 FE 0D 20 F9 18 :E4  
 :2D90=04 DD CB 01 BE CD B5 28 :15  
 :2D98=CD B3 2D C3 0D 28 CD C2 :34  
 :2DA0=2F CD F7 07 3E 01 C3 7F :7B  
 :2DA8=19 CD 25 28 80 32 0F 00 :F4  
 :2DB0=CD 8A 29 2A 4D FB CD 16 :D5  
 :2DB8=28 C3 73 28 CD FD 2D 38 :B5  
 :2DC0=05 CD F4 2D B7 C9 CD CB :0B  
 :2DC8=2D B7 C9 ED 5B B4 14 2A :E7  
 :2DD0=B0 14 CD 44 2E 28 0E 2A :63  
 :2DD8=B6 14 CD 44 2E 28 0C CD :0A  
 :2DE0=A9 2D C3 02 2A CD B6 2D :75  
 :2DE8=C3 16 28 3A 17 00 32 0F :93  
 :2DF0=00 CD 8A 29 CD B3 2D C3 :F0  
 :2DF8=02 2A 2A B4 14 D5 ED 5B :3B

:2E00=4D FB CD 44 2E 38 13 3A :0C  
 :2E08=46 FB B7 20 09 ED 5B 4F :B8  
 :2E10=FB CD 44 2E 30 04 D1 F6 :35  
 :2E18=FF C9 D1 C3 3F 1C D5 E5 :71  
 :2E20=CD 72 2A 22 28 FB CD 9E :19  
 :2E28=2A 22 2A FB E1 D1 C9 D5 :C1  
 :2E30=ED 5B 28 FB CD 44 2E 38 :E2  
 :2E38=E1 ED 5B 2A FB CD 44 2E :8D  
 :2E40=30 D8 18 D2 E5 B7 ED 52 :CD  
 :2E48=E1 C9 F5 3E E6 CD FE 0D :9B  
 :2E50=CD 49 0B 32 2D FB CD 49 :91  
 :2E58=0B 32 2E FB FE 03 28 03 :92  
 :2E60=F1 B7 C9 F1 37 C9 1C 00 :7E  
 :2E68=00 1D 00 00 CD 0F 2D CD :F3  
 :2E70=5E 2F C3 95 2D CD 0F 2D :1B  
 :2E78=CD 44 2F C3 95 2D CD 0F :A1  
 :2E80=2D CD 87 2E C3 95 2D 21 :55  
 :2E88=00 02 11 42 30 CD 7E 2B :FB  
 :2E90=CD 0B 00 11 52 30 CD 0B :43  
 :2E98=00 11 00 FF CD 03 00 D8 :B8  
 :2EA0=11 00 FF CD 8B 13 3E 04 :BD  
 :2EA8=32 80 14 21 00 FF 01 20 :07  
 :2EB0=00 CD 41 00 CD C2 2F D8 :A4  
 :2EB8=CD 4E 13 28 0E 3E 05 CD :74  
 :2EC0=EC 0D D8 11 6A 14 CD 21 :4E  
 :2EC8=13 18 E0 11 62 14 CD 21 :80  
 :2ED0=13 2A B4 14 22 32 FB 21 :75  
 :2ED8=00 01 22 42 FB 21 5B FB :D7  
 :2EE0=01 02 01 CD 44 00 CD C2 :A4  
 :2EE8=2F 38 1F 7E 23 A6 23 3C :2C  
 :2EF0=CC 0F 2F CD 29 2F 38 12 :79  
 :2EF8=21 5B FB 7E 23 A6 3C 20 :1A  
 :2F00=DC 2A 32 FB 22 B4 14 C3 :E0  
 :2F08=C2 2F CD F7 07 18 F2 3E :04  
 :2F10=1A 21 5D FB 01 00 01 ED :82  
 :2F18=B1 28 02 37 C9 21 00 01 :FD  
 :2F20=B7 ED 42 2B 22 42 FB B7 :27  
 :2F28=C9 21 5D FB 22 57 FB CD :83  
 :2F30=5F 30 DD CB 00 4E 37 C0 :7C  
 :2F38=2A B4 14 ED 5B 42 FB 19 :90  
 :2F40=22 B4 14 C9 CD 5A 25 38 :37  
 :2F48=0E 13 ED 53 57 FB 2B 22 :00  
 :2F50=59 FB 11 4C 30 18 17 F1 :01  
 :2F58=11 3D 2D C3 6B 2D 2A B0 :B0  
 :2F60=14 22 57 FB 2A B6 14 2B :A7  
 :2F68=22 59 FB 11 47 30 21 00 :1F  
 :2F70=02 CD 7E 2B CD 0B 00 11 :61  
 :2F78=52 30 CD 0B 00 11 00 FF :6A  
 :2F80=CD 03 00 D8 ED 5B 57 FB :42  
 :2F88=2A 59 FB CD C1 2A C8 D8 :D6  
 :2F90=EB ED 43 42 FB E5 C5 21 :23  
 :2F98=00 00 22 92 14 22 94 14 :92  
 :2FA0=22 96 14 11 00 FF CD 8B :34  
 :2FA8=13 21 80 14 36 04 11 5A :6D  
 :2FB0=14 CD 21 13 01 20 00 CD :03  
 :2FB8=3B 00 CD C2 2F C1 E1 D4 :6F  
 :2FC0=CA 2F F5 3E 01 CD EC 0D :F3  
 :2FC8=F1 C9 21 00 00 22 5B FB :53  
 :2FD0=21 5D FB 11 5E FB 01 FF :E3  
 :2FD8=00 36 1A ED B0 CD 12 30 :FC  
 :2FE0=38 0F CD 09 30 CD C2 2F :0B  
 :2FE8=2A 5B FB 23 22 5B FB 18 :33  
 :2FF0=DF 21 FF FF 22 5B FB 21 :97  
 :2FF8=5C FC 06 00 7E FE 1A 20 :14  
 :3000=05 36 00 2B 10 F6 23 36 :C5  
 :3008=1A 21 5B FB 01 02 01 C3 :58  
 :3010=3E 00 2A 42 FB 01 00 01 :A7  
 :3018=B7 ED 42 38 15 22 42 FB :92  
 :3020=11 5D FB 2A 57 FB E5 C5 :8F  
 :3028=ED B0 C1 E1 09 22 57 FB :BC

つづく



## リスト4-13

つづき

```

:3030=B7 C9 09 22 42 FB 44 4D :79
:3038=2A 57 FB 11 5D FB ED B0 :82
:3040=37 C9 52 65 61 64 00 53 :CF
:3048=61 76 65 00 57 72 69 74 :E2
:3050=65 00 20 46 69 6C 65 6E :73
:3058=61 6D 65 20 3F 3A 00 C5 :91
:3060=D5 E5 F5 CD A0 27 DD CB :EB
:3068=00 4E 20 0D ED 4B 42 FB :F0
:3070=ED 5B B4 14 2A 57 FB ED :79
:3078=B0 F1 E1 D1 C1 C9 CD 04 :AE
:3080=28 CD 8A 29 CD 16 28 CD :80
:3088=F4 27 11 CA 30 CD B4 30 :D7
:3090=CD A3 38 FE 0D 28 11 FE :EA
:3098=0A 20 F5 11 3B 33 CD B4 :1F
:30A0=30 CD A3 38 FE 0D 20 F9 :FC
:30A8=CD 16 28 2A 4D FB CD 73 :BD
:30B0=28 C3 0D 28 1A 13 B7 C8 :CC
:30B8=FE 1C 38 09 FE 20 30 05 :AE
:30C0=CD C8 04 18 EF CD 13 00 :80
:30C8=18 EA 09 09 3C 3C 20 43 :EF
:30D0=75 72 73 6F 72 20 4D 6F :17
:30D8=76 65 6D 65 6E 74 73 20 :22
:30E0=3E 3E 0D 0D 09 5E 53 2C :7C
:30E8=1D 09 20 4C 65 66 74 20 :F1
:30F0=43 68 61 72 61 63 74 65 :1B
:30F8=72 09 09 5E 44 2C 1C 09 :77
:3100=20 52 69 67 68 74 20 43 :81
:3108=68 61 72 61 63 74 65 72 :4A
:3110=0D 09 5E 41 09 20 4C 65 :8F
:3118=66 74 20 57 6F 72 64 09 :9F
:3120=09 5E 46 09 20 52 69 67 :F8
:3128=68 74 20 57 6F 72 64 0D :A5
:3130=09 5E 45 2C 1E 09 20 55 :74
:3138=70 20 31 20 4C 69 6E 65 :69
:3140=09 09 5E 58 2C 1F 09 20 :3C
:3148=44 6F 77 6E 20 31 20 4C :55
:3150=69 6E 65 0D 09 5E 52 09 :0B
:3158=20 46 69 6C 65 20 55 70 :85
:3160=20 53 63 72 65 65 6E 09 :89
:3168=09 5E 43 09 20 46 69 6C :EE
:3170=65 20 44 6F 77 6E 20 53 :90
:3178=63 72 65 65 6E 0D 09 5E :81
:3180=57 09 20 46 69 6C 65 20 :20
:3188=55 70 20 31 20 4C 69 6E :59
:3190=65 09 09 5E 5A 09 20 46 :9E
:3198=69 6C 65 20 44 6F 77 6E :F2
:31A0=20 31 20 4C 69 6E 65 0D :06
:31A8=09 5E 51 52 2C 46 38 09 :BD
:31B0=20 54 6F 20 54 6F 70 20 :56
:31B8=6F 66 20 46 69 6C 65 09 :7E
:31C0=09 5E 51 43 2C 46 39 09 :AF
:31C8=20 54 6F 20 45 6E 64 20 :3A
:31D0=6F 66 20 46 69 6C 65 0D :82
:31D8=09 5E 51 42 2C 46 36 09 :AB
:31E0=20 42 65 67 69 6E 20 4D :72
:31E8=61 72 6B 65 72 09 09 5E :85
:31F0=51 4B 2C 46 37 09 20 45 :B3
:31F8=6E 64 20 4D 61 72 6B 65 :E2
:3200=72 0D 09 5E 49 2C 48 54 :F7
:3208=41 42 09 20 54 41 42 20 :A3
:3210=53 65 74 09 09 5E 48 2C :10
:3218=44 45 4C 09 20 4C 65 66 :15
:3220=74 20 43 68 61 72 61 63 :D6
:3228=74 65 72 0D 0D 09 09 3C :B3
:3230=3C 20 49 6E 73 65 72 74 :D1
:3238=20 26 20 44 65 6C 65 74 :54
:3240=65 20 3E 3E 0D 0D 09 5E :82
:3248=56 09 20 49 6E 73 65 72 :80
:3250=74 20 4D 6F 64 65 20 4F :88
:3258=6E 2F 4F 66 66 0D 09 5E :2C
:3260=4E 09 20 49 6E 73 65 72 :78
:3268=74 20 61 20 52 65 74 75 :B5
:3270=72 6E 0D 09 5E 47 09 20 :C4
:3278=44 65 6C 65 74 65 20 43 :B6
:3280=68 61 72 61 63 74 65 72 :4A
:3288=20 75 6E 64 65 72 20 43 :A1
:3290=75 72 73 6F 72 0D 09 46 :97
:3298=34 09 20 44 65 6C 65 74 :4B
:32A0=65 20 43 68 61 72 61 63 :C7
:32A8=74 65 72 20 62 65 66 6F :07
:32B0=72 65 20 43 75 72 73 6F :03
:32B8=72 0D 09 5E 54 09 20 44 :A7
:32C0=65 6C 65 74 65 20 74 6F :12
:32C8=20 57 6F 72 64 20 52 69 :97
:32D0=67 68 74 0D 09 5E 59 09 :19
:32D8=20 44 65 6C 65 74 65 20 :93
:32E0=4C 69 6E 65 0D 09 5E 51 :4D
:32E8=59 2C 46 31 30 09 20 44 :99
:32F0=65 6C 65 74 65 20 74 6F :12
:32F8=20 45 6E 64 20 6F 66 20 :4C
:3300=4C 69 6E 65 0D 0D 09 28 :D3
:3308=54 79 70 65 20 5E 4A 20 :8A
:3310=46 6F 72 20 4E 65 78 74 :E6
:3318=20 46 72 61 6D 65 20 6F :9A
:3320=72 20 52 65 74 75 72 6E :12
:3328=20 74 6F 20 4F 72 69 67 :B4
:3330=69 6E 61 6C 20 46 69 6C :DF
:3338=65 29 00 0D 0D 09 09 3C :F6
:3340=3C 20 46 69 6E 64 20 26 :23
:3348=20 52 65 70 6C 61 63 65 :DC
:3350=20 3E 3E 0D 0D 09 5E 51 :6E
:3358=46 09 20 46 69 6E 64 0D :FD
:3360=09 5E 51 41 09 20 46 69 :D1
:3368=6E 64 20 26 20 52 65 70 :5F
:3370=6C 61 63 65 0D 09 5E 4C :55
:3378=09 20 46 69 6E 64 2F 52 :2B
:3380=65 70 6C 61 63 65 20 41 :CB
:3388=67 61 69 6E 0D 0D 09 09 :CB
:3390=3C 3C 20 42 6C 6F 63 6B :83
:3398=20 4F 70 65 72 61 74 69 :F4
:33A0=6F 6E 20 3E 3E 0D 0D 09 :9C
:33A8=5E 4B 42 2C 46 31 09 20 :B7
:33B0=42 65 67 69 6E 20 42 6C :B3
:33B8=6F 63 6B 09 09 5E 4B 4B :43
:33C0=2C 46 32 09 20 45 6E 64 :E4
:33C8=20 42 6C 6F 63 6B 0D 09 :21
:33D0=5E 4B 43 09 20 43 6F 70 :37
:33D8=79 20 42 6C 6F 63 6B 09 :8D
:33E0=09 5E 4B 56 09 20 4D 6F :ED
:33E8=76 65 20 42 6C 6F 63 6B :E6
:33F0=0D 09 5E 4B 59 09 20 44 :85
:33F8=65 6C 65 74 65 20 42 6C :DD
:3400=6F 63 6B 09 09 5E 4B 48 :40
:3408=2C 46 33 09 20 48 69 64 :E3
:3410=65 20 4D 61 72 6B 65 72 :E7
:3418=0D 09 5E 4B 50 09 20 50 :88
:3420=72 69 6E 74 20 42 6C 6F :FA
:3428=63 6B 09 09 5E 4B 53 09 :E5
:3430=20 53 61 76 65 20 54 65 :88
:3438=78 74 0D 09 5E 4B 52 09 :06
:3440=20 52 65 61 64 20 42 6C :6A
:3448=6F 63 6B 20 54 65 78 74 :02
:3450=09 5E 4B 57 09 20 57 72 :FB
:3458=69 74 65 20 42 6C 6F 63 :E2
:3460=6B 20 54 65 78 74 0D 09 :46
:3468=5E 4B 51 09 20 41 62 61 :27
:3470=6E 64 6F 6E 20 61 20 46 :96
:3478=69 6C 65 0D 0D 09 09 3C :A2
:3480=3C 20 4F 74 68 65 72 20 :7E
:3488=3E 3E 0D 0D 09 5E 4A 09 :50

```

つづく



リスト4-13

つづき

:3490=20 44 69 73 70 6C 61 79 :F6  
 :3498=20 48 65 6C 70 20 46 69 :78  
 :34A0=6C 65 0D 09 5E 50 09 20 :BE  
 :34A8=50 72 69 6E 74 20 54 65 :E6  
 :34B0=78 74 0D 09 5E 51 4C 09 :06  
 :34B8=20 4C 69 6E 65 20 53 65 :80  
 :34C0=61 72 63 68 0D 09 5E 51 :63  
 :34C8=4D 2C 46 35 09 20 45 78 :DA  
 :34D0=69 74 0D 0D 09 28 54 79 :F5  
 :34D8=70 65 20 52 65 74 75 72 :07  
 :34E0=6E 20 74 6F 20 4F 72 69 :BB  
 :34E8=67 69 6E 61 6C 20 46 69 :DA  
 :34F0=6C 65 29 00 ED 73 75 FC :CB  
 :34F8=3E 50 32 06 00 AF 32 72 :19  
 :3500=14 21 AB 36 22 19 00 21 :72  
 :3508=DE 36 22 39 00 3E DF 32 :BE  
 :3510=6A FC 32 6D FC 21 54 35 :AB  
 :3518=22 7E 14 21 27 35 22 0E :61  
 :3520=FA 21 00 FF 22 0C FA 31 :73  
 :3528=00 FA 01 2A 35 C5 CD A3 :8F  
 :3530=04 3E 2A CD 13 00 CD FD :16  
 :3538=10 30 FB 1A FE 2A C0 13 :50  
 :3540=CD 50 14 13 D9 21 72 35 :E5  
 :3548=06 0C BE 23 28 16 23 23 :77  
 :3550=10 F8 D9 C9 11 6A 35 CD :27  
 :3558=A3 04 CD 0B 00 3E 01 CD :8B  
 :3560=EC 0D 18 C3 5E 23 56 D5 :80  
 :3568=D9 C9 45 72 72 6F 72 20 :CC  
 :3570=21 00 44 8B 11 4D 1D 12 :7D  
 :3578=50 61 10 46 53 12 52 96 :54  
 :3580=35 53 6A 10 4C 9A 10 56 :4E  
 :3588=E1 10 54 BE 12 47 3D 36 :CF  
 :3590=58 38 37 48 9B 35 ED 7B :47  
 :3598=75 FC C9 CD 1F 11 D8 22 :31  
 :35A0=71 FC 1A 13 FE 20 28 FA :DA  
 :35A8=32 70 FC CD 1F 11 D8 22 :95  
 :35B0=73 FC 3A 70 FC 2E 00 FE :41  
 :35B8=2B 28 0E 2C FE 2D 28 09 :E9  
 :35C0=2C FE 2A 28 04 2C FE 2F :D9  
 :35C8=C0 26 00 29 11 E1 35 19 :4F  
 :35D0=7E 23 66 6F 11 E9 35 D5 :7A  
 :35D8=ED 5B 71 FC ED 4B 73 FC :5C  
 :35E0=E9 0B 36 0E 36 13 36 26 :DD  
 :35E8=36 3A 72 14 F5 AF 32 72 :3E  
 :35F0=14 3A 70 FC FE 2F 20 0B :12  
 :35F8=E5 EB CD 02 12 3E 2C CD :E8  
 :3600=13 00 E1 CD 02 12 F1 32 :F8  
 :3608=72 14 C9 EB 09 C9 EB B7 :AE  
 :3610=ED 42 C9 21 00 00 79 B0 :42  
 :3618=C8 CB 38 CB 19 30 01 19 :F9  
 :3620=CB 23 CB 12 18 F0 21 00 :F4  
 :3628=00 3E 10 CB 23 CB 12 ED :06  
 :3630=6A ED 42 38 03 1C 18 01 :09  
 :3638=09 3D 20 EF C9 1A FE 20 :56  
 :3640=2A 0E FA 28 05 CD 1F 11 :5C  
 :3648=38 27 22 5F FC CD 1F 11 :D9  
 :3650=38 19 E5 CD 1F 11 38 09 :74  
 :3658=22 6E FC 7E 32 6D FC 36 :DB  
 :3660=DF E1 22 6B FC 7E 32 6A :63  
 :3668=FC 36 DF 2A 5F FC 22 0E :C6  
 :3670=FA F3 31 00 FA ED 5B 0E :6E  
 :3678=FA 2A 0C FA 2B 36 36 2B :EC  
 :3680=36 97 2B 72 2B 73 22 0C :36  
 :3688=FA F1 C1 D1 E1 DD E1 FD :19  
 :3690=E1 ED 7B 0C FA FB C9 F3 :06  
 :3698=ED 73 0C FA 31 0C FA FD :9A  
 :36A0=E5 DD E5 E5 D5 C5 F5 FB :16  
 :36A8=C3 27 35 E5 D5 F5 21 06 :F5  
 :36B0=00 39 5E 23 56 1B 3A 6A :CF  
 :36B8=FC FE DF 28 17 2A 6B FC :A9

:36C0=B7 ED 52 28 15 3A 6D FC :D6  
 :36C8=FE DF 28 08 2A 6E FC B7 :58  
 :36D0=ED 52 28 06 F1 D1 E1 C3 :D3  
 :36D8=27 35 F3 F1 D1 E1 F3 ED :D2  
 :36E0=73 0C FA 31 0C FA FD E5 :92  
 :36E8=DD E5 E5 D5 C5 F5 2A 0C :6C  
 :36F0=FA 5E 23 56 23 22 0C FA :1C  
 :36F8=1B ED 53 0E FA FB 3A 6A :02  
 :3700=FC FE DF 28 0F 2A 6B FC :A1  
 :3708=77 3A 6D FC FE DF 28 04 :23  
 :3710=2A 6E FC 77 AF 32 72 14 :72  
 :3718=21 30 37 CD 9A 38 2A 0E :5F  
 :3720=FA 22 5D FC CD 02 12 CD :23  
 :3728=A7 04 CD DE 37 C3 27 35 :AC  
 :3730=0D 42 72 65 61 6B 20 00 :12  
 :3738=3A 72 14 F5 AF 32 72 14 :1C  
 :3740=CD 48 37 F1 32 72 14 C9 :BE  
 :3748=1A B7 CA DE 37 CD 51 14 :E2  
 :3750=2E 20 67 13 1A B7 28 08 :C9  
 :3758=CD 51 14 6F 13 1A B7 C0 :45  
 :3760=EB 01 00 10 21 5F 38 7E :32  
 :3768=BA 23 20 04 7E BB 28 06 :68  
 :3770=23 23 0C 10 F2 C9 2B CD :15  
 :3778=A7 04 CD 9A 38 3E 3D CD :92  
 :3780=13 00 06 00 79 FE 08 30 :C8  
 :3788=12 21 00 FA 09 7E CD 07 :88  
 :3790=12 E5 CD C3 37 38 22 7D :95  
 :3798=E1 77 C9 D6 08 87 4F 21 :F6  
 :37A0=00 FA 09 23 7E CD 07 12 :8A  
 :37A8=2B E5 7E CD 07 12 CD BB :FC  
 :37B0=37 38 06 EB E1 73 23 72 :49  
 :37B8=C9 E1 C9 3E 1D CD 13 00 :AE  
 :37C0=CD 13 00 3E 1D CD 13 00 :1B  
 :37C8=CD 13 00 11 00 FF CD 03 :C0  
 :37D0=00 D8 1A B7 37 C8 13 FE :B9  
 :37D8=3D 20 F7 C3 1F 11 06 0E :5B  
 :37E0=CD 94 38 21 56 38 CD 9A :AF  
 :37E8=38 CD 46 14 11 62 38 CD :D7  
 :37F0=33 38 2A 00 FA E5 7C CD :BD  
 :37F8=07 12 06 03 CD 94 38 11 :CC  
 :3800=5F 38 CD 33 38 E1 7D CD :FA  
 :3808=07 12 3E 28 CD 20 14 06 :86  
 :3810=08 26 18 29 7C CD 20 14 :EC  
 :3818=10 F7 3E 29 CD 20 14 CD :3C  
 :3820=46 14 11 7A 38 21 02 FA :3A  
 :3828=06 03 CD 3E 38 06 04 CD :23  
 :3830=3E 38 C9 EB CD 9A 38 EB :B4  
 :3838=13 3E 3D C3 20 14 CD 33 :85  
 :3840=38 D5 5E 23 56 23 EB CD :BF  
 :3848=02 12 EB D1 10 03 C3 46 :EC  
 :3850=14 CD 8F 38 18 E8 53 5A :55  
 :3858=20 48 20 50 4E 43 00 46 :AF  
 :3860=20 00 41 20 00 43 20 00 :E4  
 :3868=42 20 00 45 20 00 44 20 :2B  
 :3870=00 4C 20 00 48 20 00 41 :15  
 :3878=46 00 42 43 00 44 45 00 :54  
 :3880=48 4C 00 49 58 00 49 59 :D7  
 :3888=00 53 50 00 50 43 00 3E :74  
 :3890=20 C3 20 14 CD 8F 38 10 :BB  
 :3898=FB C9 7E B7 C8 CD 20 14 :C2  
 :38A0=23 18 F7 3E 01 C3 1B 00 :4F  
 :38A8=E5 7C CD AF 38 E1 7D F5 :68  
 :38B0=0F 0F 0F 0F CD B8 38 F1 :EA  
 :38B8=E6 0F C6 30 FE 3A 38 02 :5D  
 :38C0=C6 07 C3 23 39 3E 03 57 :84  
 :38C8=01 10 27 CD E6 38 01 E8 :0C  
 :38D0=03 CD E6 38 01 64 00 CD :20  
 :38D8=E6 38 01 0A 00 CD E6 38 :14  
 :38E0=7D F6 30 C3 23 39 1E 30 :10  
 :38E8=B7 ED 42 1C 30 FA 09 1D :52

つづく



## リスト4-13 つづき

```

:38F0=CB 4A 28 0D 7B FE 30 20 :13
:38F8=06 CB 42 C0 1E 20 01 CB :DD
:3900=8A 7B 18 1F 7E 23 B7 C8 :5C
:3908=CD 23 39 FE 0D 20 F5 7E :C7
:3910=23 FE 0A 20 F1 18 ED 3E :7F
:3918=20 18 08 CD 17 39 10 FB :68
:3920=C9 3E 0D F5 5F 3A 72 14 :28
:3928=B7 7B C4 6A 39 CD 13 00 :79
:3930=F1 C9 3A 2F 00 CB 77 37 :9C
:3938=3F C0 3A 2E 00 FE 03 37 :9F
:3940=28 18 FE 20 28 02 B7 C9 :08
:3948=CD 5A 39 3E 01 CD 1B 00 :87
:3950=FE 03 37 28 05 FE 20 20 :A3
:3958=F2 B7 F5 AF 32 2E 00 32 :DF
:3960=36 00 32 A6 0E 32 A7 0E :03
:3968=F1 C9 FE 0D 28 16 FE 09 :0A
:3970=28 0D CD DC 12 3A 36 00 :60
:3978=FE 03 37 28 DD B7 C9 CD :8A
:3980=15 13 18 F1 CD D5 12 18 :FD
:3988=EC C5 44 4D ED 79 C1 C9 :32
:3990=C5 44 4D ED 78 C1 C9 7C :C1
:3998=60 47 7D 69 4F ED 78 12 :53
:39A0=03 13 2B 7C B5 20 F6 7C :04
:39A8=60 47 7D 69 4F C9 7A 50 :6F
:39B0=47 7B 59 4F 7E ED 79 03 :51
:39B8=23 1B 7A B3 20 F6 7A 50 :4B
:39C0=47 7B 59 4F C9 D5 F5 57 :54
:39C8=7C 60 47 7D 69 4F ED 51 :96
:39D0=03 2B 7C B5 20 F8 7C 60 :53
:39D8=47 7D 69 4F F1 D1 C9 C3 :CA
:39E0=E9 3D C3 22 3E C3 5D 3E :A7
:39E8=21 77 FC 36 00 11 78 FC :4F
:39F0=01 0F 00 ED B0 21 00 FA :C8
:39F8=36 00 11 01 FA 01 0F 00 :52
:3A00=ED B0 21 61 15 22 0E FA :5E
:3A08=21 00 FF 22 0C FA 21 00 :69
:3A10=00 22 7C FC FB C9 EB 7E :C7
:3A18=B7 C4 44 16 D8 21 00 00 :CE
:3A20=22 7E FC CD C7 3E C9 EB :22
:3A28=7E B7 C4 44 16 D8 AF 32 :0C
:3A30=84 FC 2A B6 14 23 23 22 :DC
:3A38=82 FC CD 3E 3A C9 ED 73 :EC
:3A40=77 FC 31 00 FF 21 74 3A :72
:3A48=CD 04 39 21 00 A0 22 80 :6D
:3A50=FC 21 00 00 22 7E FC CD :86
:3A58=21 3D FE 55 28 0B FE 65 :47
:3A60=C2 F0 3C 21 92 3A CD 04 :AC
:3A68=39 CD 21 3D D6 80 C2 F0 :6C
:3A70=3C C3 8B 3B 0D 4C 6F 61 :EE
:3A78=64 20 73 74 65 6C 6C 61 :09
:3A80=72 20 63 6F 6D 70 69 6C :16
:3A88=65 72 20 6F 62 6A 65 63 :FA
:3A90=74 00 0D 2A 2A 20 20 44 :59
:3A98=65 62 75 67 20 6D 6F 64 :03
:3AA0=65 20 6F 62 6A 65 63 74 :FC
:3AA8=20 28 20 6E 6F 74 20 42 :1B
:3AB0=53 41 56 45 20 29 00 CD :45
:3AB8=32 39 DA 05 3D CD 21 3D :B2
:3AC0=FE FF CA BD 3C FE 56 DA :EE
:3AC8=87 3B FE 5B D2 87 3B D6 :85
:3AD0=56 87 6F 26 00 11 FA 3A :B7
:3AD8=19 5E 23 56 EB CD 04 39 :E5
:3AE0=CD 21 3D B7 28 05 CD 23 :FF
:3AE8=39 18 F5 21 83 3B CD 04 :F6
:3AF0=39 2A 7A FC CD A8 38 C3 :49
:3AF8=B7 3A 04 3B 17 3B 29 3B :E6
:3B00=3B 3B 4D 3B 0D 0D 50 72 :DA
:3B08=6F 67 72 61 6D 20 20 6E :C4
:3B10=61 6D 65 20 3A 20 00 0D :BA
:3B18=46 75 6E 63 74 69 6F 6E :46

```

```

:3B20=20 6E 61 6D 65 20 3A 20 :3B
:3B28=00 0D 43 6F 6E 73 74 61 :75
:3B30=6E 74 20 6E 61 6D 65 20 :C3
:3B38=3A 20 00 0D 56 61 72 69 :F9
:3B40=61 62 6C 65 20 6E 61 6D :F0
:3B48=65 20 3A 20 00 0D 44 61 :91
:3B50=74 61 20 20 20 20 20 6E :E3
:3B58=61 6D 65 20 3A 20 00 0D :BA
:3B60=45 6E 64 20 61 64 64 72 :D2
:3B68=65 73 73 20 20 20 3A 20 :05
:3B70=00 0D 50 72 6F 67 72 61 :78
:3B78=6D 20 73 69 7A 65 20 20 :88
:3B80=3A 20 00 20 3D 20 00 FE :D5
:3B88=80 20 18 F5 CD 21 3D E5 :BD
:3B90=CD 21 3D E1 67 22 7A FC :0B
:3B98=F1 B7 C2 B7 3A 22 7C FC :F5
:3BA0=C3 B7 3A FE 81 20 1E CD :3E
:3BA8=21 3D E5 CD 21 3D D1 57 :96
:3BB0=ED 4B 7A FC EB CD 4E 3C :F0
:3BB8=5E 23 56 70 2B 71 7A B3 :10
:3BC0=20 F2 C3 B7 3A FE 66 20 :4A
:3BC8=1E 3E CD CD 41 3C 3E DF :90
:3BD0=CD 41 3C 3E 39 CD 41 3C :0B
:3BD8=CD 21 3D CD 41 3C CD 21 :63
:3BE0=3D CD 41 3C C3 B7 3A FE :39
:3BE8=67 20 1D 3E CD CD 41 3C :F9
:3BF0=3E E2 CD 41 3C 3E 39 CD :AE
:3BF8=41 3C CD 21 3D F5 CD 41 :AB
:3C00=3C F1 B7 20 F5 C3 B7 3A :AD
:3C08=FE 68 20 1E 3E CD CD 41 :BD
:3C10=3C 3E E5 CD 41 3C 3E 39 :20
:3C18=CD 41 3C CD 21 3D CD 41 :83
:3C20=3C CD 21 3D CD 41 3C C3 :74
:3C28=B7 3A FE 41 D2 F0 3C 47 :75
:3C30=B7 20 01 04 C5 CD 21 3D :CC
:3C38=CD 41 3C C1 10 F6 C3 B7 :8B
:3C40=3A 2A 7A FC 23 22 7A FC :95
:3C48=2B CD 4E 3C 77 C9 EB 2A :D7
:3C50=82 FC 2B B7 ED 52 D4 67 :DA
:3C58=3C D2 05 3D 21 FF F8 B7 :1F
:3C60=ED 52 DA 05 3D EB C9 F5 :04
:3C68=3A 84 FC B7 20 34 E5 2A :D4
:3C70=B0 14 B7 ED 52 E1 D2 05 :72
:3C78=3D D5 11 A4 3C CD 0B 00 :DB
:3C80=D1 CD A3 38 CD 13 00 CD :26
:3C88=51 14 FE 59 C2 05 3D CD :8D
:3C90=6C 19 E5 2A B6 14 22 82 :02
:3C98=FC 3E 01 32 84 FC E1 F1 :BF
:3CA0=37 C9 F1 C9 0D 44 65 73 :E3
:3CA8=74 72 6F 79 20 61 20 46 :B5
:3CB0=49 6C 65 20 28 59 2F 4E :38
:3CB8=29 20 3F 20 00 21 5F 3B :63
:3CC0=CD 04 39 2A 7A FC 2B CD :A2
:3CC8=A8 38 21 71 3B CD 04 39 :B7
:3CD0=2A 7A FC ED 5B 7C FC B7 :17
:3CD8=ED 52 CD A8 38 2A 7C FC :8E
:3CE0=18 03 21 00 00 22 7E FC :D8
:3CE8=CD 21 39 ED 7B 77 FC C9 :CB
:3CF0=21 F8 3C CD 04 39 18 EA :61
:3CF8=0D 0A 42 61 64 20 6F 62 :0F
:3D00=6A 65 63 74 00 D5 CD 21 :69
:3D08=39 E1 CD A8 38 21 15 3D :3A
:3D10=CD 04 39 18 CD 20 4C 6F :CA
:3D18=61 64 20 65 72 72 6F 72 :0F
:3D20=00 2A 80 FC 7C E6 00 C2 :CA
:3D28=F0 3C CD 90 39 23 22 80 :87
:3D30=FC 6F C9 EB CD BE 3E 28 :10
:3D38=1A FE 26 37 C0 7E 23 CD :A3
:3D40=51 14 FE 48 37 C0 EB CD :5A
:3D48=1F 11 D8 22 7E FC EB CD :5C

```

つづく



リスト4-13

つづき

:3D50=BE 3E C0 EB 2A 7E FC 7C :C7  
 :3D58=B5 20 02 EB C9 D5 ED 73 :C0  
 :3D60=77 FC 31 00 FF AF 32 79 :FD  
 :3D68=FC 11 6E 3D D5 E9 AF 32 :57  
 :3D70=79 FC 21 7E 3D CD 04 39 :5B  
 :3D78=ED 7B 77 FC E1 C9 0D 2A :BC  
 :3D80=2A 20 45 6E 64 20 6F 66 :56  
 :3D88=20 65 78 65 63 75 74 69 :17  
 :3D90=6F 6E 0D 00 1A B7 C0 3A :B5  
 :3D98=79 FC B7 C8 E5 ED 73 77 :B0  
 :3DA0=FC F3 31 00 FA ED 5B 0E :70  
 :3DA8=FA 2A 0C FA 2B 72 2B 73 :65  
 :3DB0=22 0C FA F1 C1 D1 E1 DD :69  
 :3DB8=E1 FD E1 ED 7B 0C FA FB :28  
 :3DC0=C9 21 DA 3D CD 04 39 CD :D8  
 :3DC8=A3 38 CD 13 00 FE 79 28 :5A  
 :3DD0=06 FE 59 28 02 AF C9 F6 :F5  
 :3DD8=FF C9 0D 2A 2A 20 4F 4B :E3  
 :3DE0=20 28 59 2F 4E 29 20 3F :A6  
 :3DE8=00 F3 ED 73 0C FA 31 0C :96  
 :3DF0=FA FD E5 DD E5 E5 D5 C5 :1D  
 :3DF8=F5 2A 0C FA 5E 23 56 23 :1F  
 :3E00=22 0C FA ED 53 0E FA FB :6B  
 :3E08=3E 5B CD 13 00 2A 0E FA :AB  
 :3E10=5E 23 56 23 22 0E FA EB :0F  
 :3E18=CD C5 38 3E 5D CD 13 00 :45  
 :3E20=18 76 F3 ED 73 0C FA 31 :18  
 :3E28=0C FA FD E5 DD E5 E5 D5 :64  
 :3E30=C5 F5 2A 0C FA 5E 23 56 :C1  
 :3E38=23 22 0C FA ED 53 0E FA :93  
 :3E40=FB 3E 7B CD 13 00 2A 0E :CC  
 :3E48=FA 7E 23 B7 28 05 CD 13 :5F  
 :3E50=00 18 F6 22 0E FA 3E 7D :F3  
 :3E58=CD 13 00 18 3B F3 ED 73 :86  
 :3E60=0C FA 31 0C FA FD E5 DD :FC  
 :3E68=E5 E5 D5 C5 F5 2A 0C FA :89  
 :3E70=5E 23 56 23 22 0C FA ED :0F  
 :3E78=53 0E FA FB 21 AF 3E CD :31  
 :3E80=04 39 21 B9 3E CD 04 39 :5F  
 :3E88=2A 0E FA 5E 23 56 23 22 :4E  
 :3E90=0E FA EB CD C5 38 18 0C :E1  
 :3E98=CD 32 39 D2 A1 3D 21 AF :B8  
 :3EA0=3E CD 04 39 3E FF 32 79 :30  
 :3EA8=FC ED 7B 77 FC E1 C9 0D :8E  
 :3EB0=2A 2A 20 42 72 65 61 6B :59  
 :3EB8=00 20 69 6E 20 00 7E 23 :B8  
 :3EC0=B7 C8 FE 20 C0 18 F7 21 :8D  
 :3EC8=85 FC 36 00 11 86 FC 01 :4B  
 :3ED0=B0 01 ED B0 ED 73 85 FC :2F  
 :3ED8=31 00 FF 21 00 A0 22 8A :9D  
 :3EE0=FC 01 00 60 3E FF CD C5 :2C  
 :3EE8=39 21 6A 3F CD 04 39 21 :2E  
 :3EF0=00 00 22 88 FC 23 22 94 :7F  
 :3EF8=FC 2A B0 14 22 9F FC 21 :C8  
 :3F00=00 B0 22 AD FD 21 00 D9 :76  
 :3F08=22 AF FD 21 00 F9 22 B1 :BB  
 :3F10=FD CD 44 46 21 C3 3F CD :44  
 :3F18=04 39 11 C5 3F ED 4B B3 :3D  
 :3F20=FD 2A AD FD CD 19 40 11 :08  
 :3F28=D0 3F ED 4B B5 FD 2A AF :D2  
 :3F30=FD CD 19 40 21 DB 3F CD :2B  
 :3F38=04 39 2A B1 FD 2B CD A8 :B5  
 :3F40=38 21 F1 3F CD 04 39 21 :B4  
 :3F48=F3 3F CD 04 39 2A 88 FC :EA  
 :3F50=7C B5 20 08 21 0B 40 CD :92  
 :3F58=04 39 18 05 3E 03 CD C7 :2F  
 :3F60=38 21 0E 40 CD 04 39 C3 :74  
 :3F68=A7 40 0D 53 74 65 6C 6C :F8  
 :3F70=61 72 20 63 6F 6D 70 69 :0B  
 :3F78=6C 65 72 20 52 65 76 20 :B0

:3F80=31 2E 30 30 0D 20 28 20 :34  
 :3F88=58 31 20 56 65 72 73 69 :B2  
 :3F90=6F 6E 20 29 0D 43 6F 70 :55  
 :3F98=79 72 69 67 68 74 20 28 :DF  
 :3FA0=63 29 20 31 39 38 34 20 :A2  
 :3FA8=48 2E 57 61 74 61 6E 61 :D2  
 :3FB0=62 65 20 26 20 48 2E 4F :F2  
 :3FB8=68 6E 75 6B 69 20 2F 20 :8E  
 :3FC0=4D 49 41 0D 00 0D 50 72 :B3  
 :3FC8=6F 67 72 61 6D 20 20 00 :56  
 :3FD0=0D 44 61 74 61 20 20 20 :E7  
 :3FD8=20 20 00 0D 53 74 61 63 :D8  
 :3FE0=6B 20 62 6F 74 74 6F 6D :20  
 :3FE8=20 20 28 20 20 20 20 2D :15  
 :3FF0=00 29 00 0D 0D 2A 2A 20 :B7  
 :3FF8=20 45 6E 64 20 6F 66 20 :4C  
 :4000=63 6F 6D 70 69 6C 65 2C :15  
 :4008=20 20 00 4E 6F 00 20 65 :82  
 :4010=72 72 6F 72 28 73 29 0D :96  
 :4018=00 C5 E5 EB CD 04 39 D1 :70  
 :4020=E1 E5 D5 B7 ED 52 7C B5 :C2  
 :4028=F5 CD A8 38 F1 28 1D 3E :16  
 :4030=20 CD 23 39 3E 28 CD 23 :9F  
 :4038=39 E1 CD A8 38 3E 2D CD :FF  
 :4040=23 39 E1 2B CD A8 38 3E :53  
 :4048=29 C3 23 39 F1 F1 C9 21 :14  
 :4050=55 40 C3 9E 40 0D 25 4F :B7  
 :4058=62 6A 65 63 74 20 61 72 :FB  
 :4060=65 61 20 66 75 6C 6C 00 :99  
 :4068=21 6E 40 C3 9E 40 0D 25 :A2  
 :4070=53 79 6D 62 6F 6C 20 74 :0A  
 :4078=61 62 6C 65 20 6F 76 65 :FE  
 :4080=72 66 6C 6F 77 00 21 8C :D7  
 :4088=40 C3 9E 40 0D 25 42 61 :B6  
 :4090=64 20 73 6F 75 72 63 65 :15  
 :4098=20 66 69 6C 65 00 CD 04 :91  
 :40A0=39 21 C3 3F CD 04 39 ED :53  
 :40A8=7B 85 FC C9 21 B5 40 CD :A8  
 :40B0=04 39 C3 47 3F 0D 25 41 :F9  
 :40B8=62 6F 72 74 00 ED 53 90 :87  
 :40C0=FC 2B E5 21 C3 3F CD 04 :00  
 :40C8=39 E1 23 7E B7 28 39 E5 :B8  
 :40D0=FE 23 20 13 2A A9 FD 3E :62  
 :40D8=02 CD C7 38 21 E4 40 CD :E0  
 :40E0=04 39 18 E5 3A 20 00 FE :92  
 :40E8=40 20 0C E1 23 5E 23 56 :47  
 :40F0=E5 EB CD 04 39 18 D2 FE :C2  
 :40F8=5C 20 08 2A 90 FC CD 04 :0B  
 :4100=39 18 C6 CD 23 39 18 C1 :19  
 :4108=2A 88 FC 23 22 88 FC 21 :98  
 :4110=C3 3F CD 04 39 21 A7 FC :D0  
 :4118=3A 9C FC B7 28 06 CD D4 :58  
 :4120=42 21 27 FD CD 04 39 11 :A2  
 :4128=6E 41 CD 0B 00 CD A3 38 :2F  
 :4130=FE 20 C8 FE 45 28 04 FE :53  
 :4138=65 20 F2 3A 9C FC B7 20 :20  
 :4140=27 ED 5B 9D FC 2A B6 14 :FC  
 :4148=B7 ED 52 EB 28 0E 2A A7 :E8  
 :4150=FD 11 A7 FC B7 ED 52 ED :94  
 :4158=5B 9D FC 19 CD E1 28 CD :B0  
 :4160=F5 28 22 B4 14 C3 A6 14 :84  
 :4168=11 89 41 C3 0B 00 0D 3C :F2  
 :4170=73 70 61 63 65 3E 20 63 :CD  
 :4178=6F 6E 74 69 6E 75 65 2C :2E  
 :4180=20 45 28 64 69 74 29 0D :04  
 :4188=00 0D 25 49 6E 63 6C 75 :2D  
 :4190=64 65 20 73 6F 75 72 63 :15  
 :4198=65 20 69 73 20 6E 6F 74 :D2  
 :41A0=20 6F 6E 20 6D 65 6D 6F :CB  
 :41A8=72 79 00 CD 02 64 EB 06 :0F

つづく



## リスト4-13 つづき

```

:41B0=12 E5 C5 AF 77 23 10 FC :11
:41B8=C1 E1 CD C7 41 28 05 77 :1B
:41C0=23 13 10 F6 EB B7 C9 1A :C1
:41C8=B7 C8 FE 20 C8 38 1F FE :BA
:41D0=2C C8 FE 3A C8 FE 3B C8 :F5
:41D8=FE 3C C8 FE 3D C8 FE 3E :41
:41E0=C8 FE 5B C8 FE 5D C8 FE :0A
:41E8=3F 28 03 FE 2A C0 F1 37 :7A
:41F0=C9 3A 9C FC B7 C2 52 66 :CC
:41F8=2A A7 FD CD 02 64 E5 11 :F7
:4200=00 FF CD AB 41 DA EF 42 :C3
:4208=22 A7 FD E5 CD 53 61 FE :2A
:4210=3B C2 52 66 2A 7E 14 22 :93
:4218=92 FC 21 0F 43 22 7E 14 :B5
:4220=11 00 FF CD 94 13 3E 04 :C6
:4228=32 80 14 21 00 FF 01 20 :07
:4230=00 CD 41 00 CD C2 2F DA :A6
:4238=0F 43 CD 4E 13 28 10 3E :F6
:4240=05 CD EC 0D DA 0F 43 11 :08
:4248=6A 14 CD 21 13 18 DC 11 :84
:4250=62 14 CD 21 13 21 00 01 :99
:4258=22 A3 FC 22 A5 FC 3E FF :C1
:4260=32 9C FC 2A A7 FD 22 98 :52
:4268=FC 2A A9 FD 22 96 FC 2A :AA
:4270=94 FC 22 9A FC 21 01 00 :6A
:4278=22 94 FC 21 27 FD 22 A7 :C0
:4280=FD 36 00 21 9E 42 CD 04 :05
:4288=39 E1 D1 B7 ED 52 EB 7A :46
:4290=B3 C8 D5 E5 7E CD 23 39 :DC
:4298=E1 D1 23 1B 18 F1 0D 2A :30
:42A0=2A 20 20 49 6E 63 6C 75 :65
:42A8=64 65 20 3A 20 00 3A 9C :19
:42B0=FC B7 C8 AF 32 9C FC 21 :15
:42B8=DB 42 CD 04 39 CD D4 42 :0A
:42C0=2A 96 FC 22 A9 FD 2A 9A :48
:42C8=FC 22 94 FC 2A 98 FC 22 :8E
:42D0=A7 FD 37 C9 2A 92 FC 22 :7E
:42D8=7E 14 C9 0D 2A 2A 20 20 :FC
:42E0=45 6E 64 20 6F 66 20 69 :95
:42E8=6E 63 6C 75 64 65 00 21 :9C
:42F0=F8 42 CD BD 40 C3 AC 40 :B3
:42F8=25 42 61 64 20 69 6E 63 :86
:4300=6C 75 64 65 20 66 69 6C :05
:4308=65 20 6E 61 6D 65 00 21 :47
:4310=18 43 CD BD 40 C3 AC 40 :D4
:4318=25 4E 6F 20 69 6E 63 6C :A8
:4320=75 64 65 20 66 69 6C 65 :FE
:4328=00 2A 9F FC 22 9D FC EB :6B
:4330=21 27 FD 3A 9C FC B7 20 :EE
:4338=0D 1A B7 37 21 A7 FC 36 :0F
:4340=00 ED 5B 94 FC C8 06 80 :26
:4348=E5 C5 E5 3A 9C FC B7 20 :38
:4350=05 CD 86 43 18 03 CD 97 :1A
:4358=43 E1 C1 FE 1A 28 0B FE :2E
:4360=0D 28 16 77 23 10 E2 C3 :9A
:4368=86 40 36 00 ED 5B 94 FC :D4
:4370=3A 9C FC B7 28 0E 37 18 :0E
:4378=0B 36 00 2A 94 FC 54 5D :AC
:4380=23 22 94 FC E1 C9 2A 9F :48
:4388=FC 7E B7 20 04 37 3E 1A :E4
:4390=C9 23 22 9F FC B7 C9 2A :53
:4398=A3 FC ED 5B A5 FC E5 B7 :24
:43A0=ED 52 E1 38 22 21 00 00 :9B
:43A8=22 A3 FC 21 00 01 22 A5 :AA
:43B0=FC 21 5B FB 01 02 01 CD :44
:43B8=44 00 CD C2 2F DA 0F 43 :2E
:43C0=7E 23 A6 3C CC D6 43 2A :92
:43C8=A3 FC E5 11 5D FB 19 7E :84
:43D0=E1 23 22 A3 FC C9 3E 1A :E6
:43D8=21 5D FB 01 00 01 ED B1 :19

```

```

:43E0=28 02 37 C9 21 00 01 B7 :03
:43E8=ED 42 22 A5 FC B7 C9 3A :AC
:43F0=AB FD CB 67 C8 3E 68 CD :15
:43F8=FF 43 2A A9 FD 18 30 CD :27
:4400=61 44 2A B3 FD 23 23 23 :E8
:4408=22 B3 FD C9 3A AB FD CB :48
:4410=67 C8 3E FF 32 87 FC C9 :EA
:4418=3A AB FD CB 67 C8 AF 32 :BD
:4420=87 FC C9 3A 87 FC B7 C8 :88
:4428=E5 3E 66 CD FF 43 E1 E5 :5E
:4430=7D CD 61 44 E1 7C CD 61 :7A
:4438=44 2A B3 FD 23 23 22 B3 :39
:4440=FD C9 3A 87 FC B7 C8 E5 :E7
:4448=3E 67 CD FF 43 E1 7E E5 :F8
:4450=CD 61 44 2A B3 FD 23 22 :91
:4458=B3 FD E1 7E 23 B7 20 EE :F7
:4460=C9 5F 2A 8A FC 7C E6 00 :3A
:4468=C2 4F 40 7B 23 22 8A FC :97
:4470=2B C3 89 39 CB 66 28 2A :33
:4478=11 F0 F8 01 00 0C E5 D5 :C0
:4480=2A 8E FC B7 ED 52 28 44 :16
:4488=D1 E1 E5 D5 23 13 1A BE :7A
:4490=20 07 B7 28 5E 10 F5 18 :81
:4498=5A D1 21 F0 FF 19 EB E1 :20
:44A0=18 D9 11 00 D9 01 00 0C :E8
:44A8=E5 D5 2A 8C FC B7 ED 52 :62
:44B0=28 26 D1 E1 E5 D5 23 13 :F0
:44B8=1A BE 20 07 B7 28 34 10 :22
:44C0=F5 18 30 D1 21 10 00 19 :58
:44C8=EB E1 18 D9 2A 8E FC 11 :82
:44D0=F0 FF 19 22 8E FC 18 0A :D6
:44D8=2A 8C FC 11 10 00 19 22 :0E
:44E0=8C FC 2A 8E FC 11 20 00 :6D
:44E8=19 ED 5B 8C FC B7 ED 52 :DF
:44F0=CA 68 40 D1 E1 01 10 00 :35
:44F8=ED B0 C9 11 F0 F8 01 00 :60
:4500=0C E5 D5 1A B7 28 14 23 :F6
:4508=13 1A BE 20 13 B7 28 02 :FF
:4510=10 F5 E1 D1 01 10 00 ED :B5
:4518=B0 AF C9 E1 2A 8C FC FE :B9
:4520=E1 11 F0 FF 19 E5 11 00 :F0
:4528=D9 B7 ED 52 D1 E1 30 CE :7F
:4530=F6 FF C9 ED 5B 8E FC 21 :B1
:4538=F0 F8 22 8E FC B7 ED 52 :8A
:4540=44 4D 21 10 00 19 16 00 :F1
:4548=1E 01 78 B1 C8 1D 20 25 :72
:4550=1E 10 CB 7E 28 1F C5 D5 :58
:4558=E5 23 E5 11 0C 00 19 36 :59
:4560=00 21 7A 45 CD 04 39 E1 :CB
:4568=CD 04 39 2A 88 FC 23 22 :FD
:4570=88 FC E1 D1 C1 72 23 0B :97
:4578=18 D0 0D 20 20 20 20 3F :B4
:4580=3A 20 55 6E 64 65 66 69 :B5
:4588=6E 65 64 20 6C 61 62 65 :EB
:4590=6C 20 3A 20 00 21 00 D9 :E0
:4598=22 8C FC 01 00 20 36 00 :01
:45A0=23 0B 78 B1 20 F8 21 F0 :80
:45A8=F8 22 8E FC C9 11 00 D9 :57
:45B0=2A 8C FC B7 ED 52 D5 DD :5A
:45B8=E1 7C B5 28 5E E5 DD 6E :C8
:45C0=0D DD 66 0E DD 36 0D 00 :7E
:45C8=DD CB 00 7E 28 19 DD E5 :29
:45D0=21 21 46 CD 04 39 E1 E5 :58
:45D8=23 CD 04 39 2A 88 FC 23 :FE
:45E0=22 88 FC DD E1 18 29 06 :AB
:45E8=58 DD 7E 00 E6 0F 3D 28 :0D
:45F0=08 04 3D 28 04 04 3D 20 :D6
:45F8=17 C5 DD E5 CD F6 5F E1 :A1
:4600=F1 E5 23 11 CD FD 01 0D :E2
:4608=00 ED B0 CD E3 5F DD E1 :6A

```

つづく



リスト4-13

つづき

:4610=11 10 00 DD 19 E1 B7 ED :9C  
 :4618=52 18 9E 2A B3 FD C3 F6 :9B  
 :4620=5F 0D 20 20 20 20 3F 3A :65  
 :4628=20 55 6E 64 65 66 69 6E :E9  
 :4630=65 64 20 66 75 6E 63 74 :09  
 :4638=69 6F 6E 20 6E 61 6D 65 :07  
 :4640=20 3A 20 00 CD 29 43 D8 :8B  
 :4648=22 A7 FD ED 53 A9 FD CD :79  
 :4650=53 61 FE 80 C2 52 66 CD :79  
 :4658=53 61 3D C2 52 06 CD D0 :08  
 :4660=5F AF 32 AB FD CD 53 61 :69  
 :4668=FE 28 C2 52 66 CD 53 61 :21  
 :4670=FE 29 CA F1 46 FE 25 C2 :0D  
 :4678=02 65 CD 53 61 FE 84 20 :8A  
 :4680=0A 21 AB FD CB E6 CD 53 :A4  
 :4688=61 18 5A 3D C2 02 65 2A :63  
 :4690=BE FD 7C B7 C2 02 65 7D :94  
 :4698=CD F9 63 F5 CD 53 61 FE :9D  
 :46A0=3A C2 02 65 F1 FE 50 20 :C2  
 :46A8=09 21 AB FD CB FE 06 80 :21  
 :46B0=18 19 FE 44 20 09 21 AB :68  
 :46B8=FD CB F6 06 40 18 0C FE :26  
 :46C0=53 C2 02 65 21 AB FD CB :10  
 :46C8=EE 06 20 C5 CD 53 61 CD :27  
 :46D0=45 60 F1 87 30 03 22 AD :1F  
 :46D8=FD 87 30 03 22 AF FD 87 :0C  
 :46E0=30 03 22 B1 FD 3A DD FD :17  
 :46E8=FE 2C 28 81 FE 29 C2 02 :BE  
 :46F0=65 CD 53 61 FE 3B C2 52 :33  
 :46F8=66 3A AB FD CB 67 3E 55 :0D  
 :4700=28 02 3E 65 CD 61 44 2A :69  
 :4708=AD FD 22 B3 FD CD F6 5F :9E  
 :4710=2A AF FD 22 B5 FD 22 B7 :83  
 :4718=FD 3E 56 CD E3 5F 21 35 :F6  
 :4720=4A CD 04 39 21 CD FD CD :0C  
 :4728=04 39 CD 5B 4A CD 95 45 :56  
 :4730=2A AF FD CD CB 4A 5F 77 :8E  
 :4738=6F 72 6B 00 2A B5 FD CD :F5  
 :4740=CB 4A 5F 76 61 72 00 2A :E7  
 :4748=AD FD CD CB 4A 5F 63 6F :BD  
 :4750=64 65 00 16 3E 1E 01 CD :09  
 :4758=A8 5F 3E 32 2A AF FD 11 :5E  
 :4760=21 00 19 CD B2 5F 3E 01 :57  
 :4768=B7 28 22 3D 20 08 11 73 :EA  
 :4770=ED CD A8 5F 18 0D 3E 21 :45  
 :4778=21 00 00 CD B2 5F 3E 22 :5F  
 :4780=CD 97 5F 2A AF FD 11 22 :CC  
 :4788=00 19 CD AA 5F 3A AB FD :D1  
 :4790=2A B1 FD CB 6F 20 15 3E :85  
 :4798=00 21 00 F9 B7 28 0D E5 :EB  
 :47A0=11 7B ED CD A8 5F E1 CD :FB  
 :47A8=AA 5F 18 05 3E 31 CD B2 :14  
 :47B0=5F 3E 21 2A AD FD 11 03 :A6  
 :47B8=00 19 CD B2 5F 3E E5 CD :E7  
 :47C0=97 5F CD 53 61 AF 32 33 :8B  
 :47C8=FE CD E2 4A FE 7B C2 52 :84  
 :47D0=66 3E FF 32 AC FD 2A A9 :51  
 :47D8=FD CD 23 44 CD 89 51 CD :A5  
 :47E0=CC 4D CD A1 51 AF 32 AC :65  
 :47E8=FD CD 53 61 3E C9 CD 97 :E9  
 :47F0=5F CD 33 45 3A DD FD 3C :F4  
 :47F8=CA 2C 4A AF 32 33 FE 67 :B9  
 :4800=6F 22 34 FE F5 CD E2 4A :B1  
 :4808=2F 32 DE FD 3A DD FD FE :4E  
 :4810=FF 20 04 F1 C3 2C 4A FE :4B  
 :4818=8F 20 07 F1 F6 80 F5 CD :DF  
 :4820=53 61 3D C2 1B 65 21 BD :11  
 :4828=FD CD FB 44 06 09 B7 20 :EF  
 :4830=16 3A BD FD FE 89 28 07 :C0  
 :4838=C5 CD 38 65 C1 18 08 2A :3A

:4840=CA FD F5 CD FB 5F C1 2A :CE  
 :4848=B3 FD 22 CA FD CB B8 21 :3D  
 :4850=BD FD 70 CD 74 44 CD D0 :4C  
 :4858=5F 3E 57 CD E3 5F 21 48 :6C  
 :4860=4A CD 04 39 21 CD FD E5 :24  
 :4868=CD 04 39 E1 CD 42 44 CD :0B  
 :4870=53 61 FE 28 C2 52 66 2A :7E  
 :4878=B5 FD 22 2F FE AF 32 31 :13  
 :4880=FE CD 53 61 FE 3B 28 44 :24  
 :4888=FE 29 28 7B 3D C2 4A 65 :78  
 :4890=18 03 CD 53 61 CD A6 4C :5B  
 :4898=F6 02 2A B5 FD 22 CA FD :BD  
 :48A0=21 BD FD 77 CD 74 44 2A :01  
 :48A8=B5 FD 23 22 B5 FD 22 B7 :82  
 :48B0=FD 21 31 FE 7E FE 20 30 :19  
 :48B8=01 34 CD 53 61 FE 2C 28 :08  
 :48C0=D1 FE 3B 28 07 FE 29 28 :88  
 :48C8=3E C3 52 66 CD 53 61 FE :38  
 :48D0=23 20 07 F1 F6 08 F5 CD :FB  
 :48D8=53 61 FE BF 20 07 F1 F6 :7F  
 :48E0=04 F5 CD 53 61 FE 2C 20 :C4  
 :48E8=19 CD 53 61 FE 23 20 07 :E2  
 :48F0=F1 F6 02 F5 CD 53 61 FE :5D  
 :48F8=C0 20 07 F1 F6 01 F5 CD :91  
 :4900=53 61 FE 29 C2 52 66 F1 :46  
 :4908=CB 57 28 0E F5 3E 22 2A :D7  
 :4910=AF FD 11 2C 00 19 CD B2 :81  
 :4918=5F F1 CB 47 28 12 F5 11 :A2  
 :4920=53 ED CD A8 5F 2A AF FD :EA  
 :4928=11 2E 00 19 CD AA 5F F1 :1F  
 :4930=F5 3A 31 FE 5F 16 0E CD :AE  
 :4938=A8 5F 3E 11 21 00 00 CD :44  
 :4940=B2 5F 2A 2F FE F1 CB 7F :A3  
 :4948=28 08 E5 ED 5B B3 FD 1B :28  
 :4950=1B D5 F5 3E 21 CD B2 5F :22  
 :4958=3E CD 2A AD FD 11 24 00 :14  
 :4960=19 CD B2 5F F1 CB 5F 20 :32  
 :4968=0D F5 11 E5 DD CD A8 5F :A9  
 :4970=3E FF 32 33 FE F1 CB 4F :AB  
 :4978=20 0D F5 11 E5 FD CD A8 :8A  
 :4980=5F 3E FF 32 33 FE F1 CB :BB  
 :4988=57 28 12 F5 11 2A DD CD :6B  
 :4990=A8 5F 2A AF FD 11 2C 00 :1A  
 :4998=19 CD AA 5F F1 CB 47 28 :1A  
 :49A0=12 F5 11 2A FD CD A8 5F :13  
 :49A8=2A AF FD 11 2E 00 19 CD :FB  
 :49B0=AA 5F F1 F5 CD 53 61 FE :6E  
 :49B8=3B C2 52 66 CD 53 61 3E :74  
 :49C0=FF CD E2 4A FE 7B C2 52 :85  
 :49C8=66 3E FF 32 AC FD 2A A9 :51  
 :49D0=FD CD 23 44 CD 89 51 CD :A5  
 :49D8=CC 4D CD A1 51 AF 32 AC :65  
 :49E0=FD CD 53 61 2A 34 FE 7C :56  
 :49E8=B5 C4 FB 5F F1 CB 4F 20 :FE  
 :49F0=08 F5 11 E1 FD CD A8 5F :C0  
 :49F8=F1 CB 5F 20 08 F5 11 E1 :2A  
 :4A00=DD CD A8 5F F1 F5 3E C9 :9E  
 :4A08=CD 97 5F F1 87 D2 F1 47 :45  
 :4A10=E1 CD F6 5F D1 2A B5 FD :B0  
 :4A18=B7 ED 52 CD AA 5F 2A B3 :A9  
 :4A20=FD 2B 2B 22 B3 FD CD F6 :E8  
 :4A28=5F C3 F1 47 CD AD 45 3E :57  
 :4A30=FF CD 61 44 C9 0D 0A 50 :A1  
 :4A38=72 6F 67 72 61 6D 20 20 :C8  
 :4A40=6E 61 6D 65 20 3A 20 00 :1B  
 :4A48=0D 0A 46 75 6E 63 74 69 :80  
 :4A50=6F 6E 20 6E 61 6D 65 20 :BE  
 :4A58=3A 20 00 01 F7 00 21 9D :10  
 :4A60=66 DD 21 94 67 FD 21 B6 :33  
 :4A68=67 7D DD BE 00 20 1B 7C :36

つづく



## リスト4-13 つづき

```

:4A70=DD BE 01 20 15 DD 23 DD :AE
:4A78=23 11 9D 66 7E 93 5F 23 :CA
:4A80=7E 9A 57 E5 2A AD FD 19 :41
:4A88=18 29 7D FD BE 00 20 1B :B4
:4A90=7C FD BE 01 20 15 FD 23 :8D
:4A98=FD 23 11 36 FE 7E 93 5F :D5
:4AA0=23 7E 9A 57 E5 2A AF FD :4D
:4AA8=19 18 08 7E E5 C5 CD 97 :C5
:4AB0=5F 18 05 0B C5 CD AA 5F :22
:4AB8=C1 E1 23 0B 78 B1 20 A9 :C2
:4AC0=2A B5 FD 11 30 00 19 22 :58
:4AC8=B5 FD C9 22 CA FD 21 BD :42
:4AD0=FD 36 01 D1 1A 13 23 77 :CC
:4AD8=B7 20 F9 D5 21 BD FD C3 :43
:4AE0=74 44 32 DE FD 3A DD FD :D9
:4AE8=FE 25 20 08 CD 53 61 CD :99
:4AF0=59 4D 18 F1 FE 8C 20 05 :5E
:4AF8=CD 0E 4B 18 E8 FE 8D 20 :D1
:4B00=05 CD 53 4B 18 DF FE 8E :F3
:4B08=C0 CD F0 4B 18 D7 CD 53 :D7
:4B10=61 3D C2 4A 65 CD A6 4C :CE
:4B18=F6 01 01 0D 00 21 BD FD :E0
:4B20=77 CD 37 61 CD 53 61 FE :5B
:4B28=F0 C2 52 66 CD 53 61 CD :B8
:4B30=0B 60 22 CA FD 01 0D 00 :62
:4B38=21 BD FD CD 47 61 21 BD :2E
:4B40=FD CD 74 44 3A DD FD FE :94
:4B48=2C 28 C3 FE 3B C2 52 66 :CA
:4B50=C3 53 61 CD 53 61 3D C2 :F7
:4B58=4A 65 CD A6 4C F6 02 2A :90
:4B60=B5 FD 22 2F FE 21 01 00 :23
:4B68=22 31 FE 01 0D 00 21 BD :3D
:4B70=FD 77 CD 37 61 CD 53 61 :5A
:4B78=FE 5B 20 14 CD 53 61 CD :DB
:4B80=0B 60 22 31 FE 3A DD FD :D0
:4B88=FE 5D C2 52 66 CD 53 61 :56
:4B90=FE BE 20 22 CD 53 61 FE :7D
:4B98=28 C2 52 66 CD 53 61 CD :F0
:4BA0=0B 60 22 2F FE 21 00 00 :DB
:4BA8=22 31 FE 3A DD FD FE 29 :8C
:4BB0=C2 52 66 CD 53 61 2A 2F :54
:4BB8=FE 22 CA FD 01 0D 00 21 :16
:4BC0=BD FD CD 47 61 21 BD FD :0A
:4BC8=CD 74 44 2A B5 FD ED 5B :A9
:4BD0=31 FE 19 22 B5 FD 22 B7 :F5
:4BD8=FD 3A DD FD FE 2C CA 53 :58
:4BE0=4B FE 3B C2 52 66 C3 53 :14
:4BE8=61 3E 02 32 DF FD 18 18 :DF
:4BF0=3E 01 32 DF FD 3E C3 21 :6F
:4BF8=00 00 CD B2 5F 2A B3 FD :B8
:4C00=2B 2B 22 2B FE CD 53 61 :22
:4C08=3A DF FD 3D 20 27 3A DD :B1
:4C10=FD 3D 20 21 2A A7 FD 7E :C7
:4C18=FE 3A 20 19 23 22 A7 FD :5A
:4C20=CD A6 4C F6 03 2A B3 FD :92
:4C28=22 CA FD 21 BD FD 77 CD :08
:4C30=74 44 CD 53 61 3A DD FD :4D
:4C38=FE 22 20 2E 2A A7 FD 7E :BA
:4C40=23 FE 22 20 0F 7E 23 FE :11
:4C48=22 28 09 2B 22 A7 FD CD :11
:4C50=53 61 18 35 B7 20 0C 2B :0F
:4C58=22 A7 FD CD 8C 65 CD 53 :A4
:4C60=61 18 26 E5 CD 97 5F E1 :28
:4C68=18 D5 FE 23 28 04 FE 92 :CA
:4C70=20 0B CD 53 61 CD 45 60 :1E
:4C78=CD AA 5F 18 0C FE 91 CC :55
:4C80=53 61 CD 31 60 7D CD 97 :F3
:4C88=5F 3A DD FD FE 2C CA 05 :6C
:4C90=4C FE 3B C2 52 66 21 DF :FF
:4C98=FD 7E 36 00 3D 2A 2B FE :41

```

```

:4CA0=CC FB 5F C3 53 61 21 BD :7B
:4CA8=FD CD FB 44 B7 20 13 3A :2D
:4CB0=DE FD B7 20 05 CD 5E 65 :47
:4CB8=18 08 3A BD FD CB 67 C4 :0A
:4CC0=5E 65 3A DE FD B7 C8 3E :95
:4CC8=10 C9 3A DD FD 3D 20 41 :8B
:4CD0=2A A7 FD 7E FE 3A 20 39 :DD
:4CD8=23 7E FE 3D 28 33 22 A7 :00
:4CE0=FD 21 BD FD CD FB 44 B7 :9B
:4CE8=20 16 3A BD FD CB 67 28 :84
:4CF0=0F FE 98 28 05 CD 70 65 :74
:4CF8=18 06 2A CA FD CD FB 5F :36
:4D00=2A B3 FD 22 CA FD 21 BD :A1
:4D08=FD 36 18 CD 74 44 CD 53 :F0
:4D10=61 CD 75 5E 3A DD FD 21 :36
:4D18=32 4D 01 0D 00 ED B1 C2 :ED
:4D20=B5 51 3E 0C 91 87 4F 21 :D8
:4D28=3F 4D 09 5E 23 56 D5 C3 :04
:4D30=53 61 3B 25 7B 90 96 98 :4D
:4D38=9B 93 9E 9D 9C 9F A0 CB :0F
:4D40=4D 59 4D A0 4D E9 4B DA :EE
:4D48=4D 10 4E 46 4F 3B 50 BF :8A
:4D50=50 DF 50 D7 50 35 51 5A :86
:4D58=51 FE 81 20 06 CD F1 41 :F5
:4D60=C3 53 61 FE 8B 20 15 CD :02
:4D68=53 61 FE 3B C2 52 66 3A :A1
:4D70=AC FD B7 CA 52 66 CD EF :9E
:4D78=43 C3 53 61 D6 85 20 0E :43
:4D80=CD 53 61 FE 3B C2 52 66 :34
:4D88=CD 0C 44 C3 53 61 3D C2 :93
:4D90=52 66 CD 53 61 FE 3B C2 :34
:4D98=52 66 CD 18 44 C3 53 61 :58
:4DA0=CD 89 51 CD CF 4D CD 53 :B0
:4DA8=61 FE 97 20 1B CD 75 5E :D1
:4DB0=CD 53 61 CD F3 53 3E B7 :89
:4DB8=CD 97 5F 2A 21 FE 3E 28 :72
:4DC0=CD 6E 51 3E CA DC B2 5F :81
:4DC8=CD A1 51 C9 CD 53 61 CD :D6
:4DD0=CA 4C 3A DD FD FE 7D 20 :C5
:4DD8=F6 C9 CD 89 51 CD FE 53 :84
:4DE0=FE 7B C2 52 66 3E B7 CD :B5
:4DE8=97 5F 3E CA 21 00 00 CD :EC
:4DF0=B2 5F 2A B3 FD 2B 2B 22 :63
:4DF8=23 FE CD CC 4D 2A 21 FE :50
:4E00=3E 18 CD 6E 51 3E C3 DC :BF
:4E08=B2 5F CD A1 51 C3 53 61 :47
:4E10=3D C2 52 66 21 BD FD CD :5F
:4E18=FB 44 B7 C4 B8 64 21 BD :B4
:4E20=FD 7E E6 2F FE 02 C2 52 :A4
:4E28=66 01 0F 00 CD 37 61 21 :FC
:4E30=BD FD CB EE CD 74 44 01 :F9
:4E38=06 00 21 25 FE CD 37 61 :AF
:4E40=2A CA FD 22 25 FE CD 53 :56
:4E48=61 FE F0 C2 52 66 CD 53 :E9
:4E50=61 CD FE 53 FE 99 C2 52 :2A
:4E58=66 3E 32 2A 25 FE CD B2 :A2
:4E60=5F CD 53 61 CD FE 53 2A :28
:4E68=B7 FD E5 22 27 FE CD 21 :CE
:4E70=4F 3E 32 2B CD B2 5F 21 :E9
:4E78=00 00 22 29 FE 3A DD FD :5D
:4E80=FE 7B 28 1F FE 9A C2 52 :6C
:4E88=66 CD 53 61 CD FE 53 FE :03
:4E90=7B C2 52 66 2A B7 FD 22 :F5
:4E98=29 FE CD 21 4F 3E 32 2B :FF
:4EA0=CD B2 5F CD 89 51 2A 27 :D6
:4EA8=FE CD 81 5E 2A 25 FE CD :C4
:4EB0=81 5E 21 32 4F CD DE 5E :8A
:4EB8=2A B3 FD 2B 2B 22 23 FE :73
:4EC0=CD CC 4D 2A 29 FE 7C B5 :68
:4EC8=20 19 2A 25 FE 3E 21 CD :B2

```

つづく



リスト4-13

つづき

:4ED0=B2 5F 3E 34 CD 97 5F 2A :70  
 :4ED8=21 FE 3E 20 CD 6E 51 3E :47  
 :4EE0=C2 18 19 CD 81 5E 2A 25 :EE  
 :4EE8=FE CD 81 5E 21 3D 4F CD :24  
 :4EF0=DE 5E 2A 21 FE 3E 30 CD :C0  
 :4EF8=6E 51 3E D2 DC B2 5F CD :89  
 :4F00=A1 51 E1 22 B7 FD 01 06 :B0  
 :4F08=00 21 25 FE CD 47 61 01 :BA  
 :4F10=0F 00 21 BD FD CD 47 61 :5F  
 :4F18=21 BD FD CD 74 44 C3 53 :76  
 :4F20=61 23 22 B7 FD EB 2A B5 :24  
 :4F28=FD B7 ED 52 EB D0 22 B5 :85  
 :4F30=FD C9 0A 3A B9 B9 21 BB :58  
 :4F38=BB 96 DA 00 00 08 3A B9 :26  
 :4F40=B9 21 BB BB 86 77 FE 23 :6E  
 :4F48=CA D0 4F 3D C2 52 66 21 :C1  
 :4F50=BD FD CD FB 44 B7 C4 B8 :F9  
 :4F58=64 21 BD FD 7E E6 2F FE :D0  
 :4F60=02 C2 52 66 01 0F 00 CD :59  
 :4F68=37 61 21 BD FD CB EE CD :F9  
 :4F70=74 44 2A 25 FE E5 2A CA :DE  
 :4F78=FD 22 25 FE CD 53 61 FE :C1  
 :4F80=2C C2 52 66 CD 53 61 CD :F4  
 :4F88=FE 53 FE 7B C2 52 66 3E :82  
 :4F90=32 2A 25 FE CD B2 5F CD :2A  
 :4F98=89 51 CD CC 4D 3E 21 2A :49  
 :4FA0=25 FE CD B2 5F 3E 35 CD :41  
 :4FA8=97 5F 2A 21 FE 3E 20 CD :6A  
 :4FB0=6E 51 3E C2 DC B2 5F CD :79  
 :4FB8=A1 51 E1 22 25 FE 01 0F :28  
 :4FC0=00 21 BD FD CD 47 61 21 :71  
 :4FC8=BD FD CD 74 44 C3 53 61 :B6  
 :4FD0=21 E0 FD 7E B7 C2 52 66 :AD  
 :4FD8=2F 77 CD 53 61 FE 2C C2 :13  
 :4FE0=52 66 CD 53 61 CD 39 54 :93  
 :4FE8=47 3A E2 FD 3D C2 81 66 :46  
 :4FF0=78 FE 7B C2 52 66 FD 7E :E6  
 :4FF8=FD B7 28 11 21 30 54 3D :CF  
 :5000=28 08 21 38 50 CD D3 5E :D7  
 :5008=18 08 CD D3 5E 3E 47 CD :70  
 :5010=97 5F CD 89 51 CD CC 4D :83  
 :5018=2A 21 FE 3E 10 CD 6E 51 :23  
 :5020=30 0C E5 3E 05 CD 97 5F :27  
 :5028=E1 3E C2 CD B2 5F AF 32 :A0  
 :5030=E0 FD CD A1 51 C3 53 61 :13  
 :5038=02 06 B8 2A 2D FE E5 2A :24  
 :5040=2B FE E5 21 00 00 22 2D :7E  
 :5048=FE 22 2B FE CD FE 53 FE :65  
 :5050=94 C2 52 66 3E B7 CD 97 :67  
 :5058=5F 3E CA 21 00 00 CD B2 :07  
 :5060=5F 2A B3 FD 2B 2B 22 2B :DC  
 :5068=FE CD 53 61 CD CA 4C 3A :9C  
 :5070=DD FD FE BD 28 2A FE 95 :7A  
 :5078=28 08 2A 2B FE CD FB 5F :AA  
 :5080=18 0D CD 53 61 FE 93 28 :5F  
 :5088=17 CD A8 50 CD CA 4C 2A :E9  
 :5090=2D FE 7C B5 C4 FB 5F E1 :5B  
 :5098=22 2B FE E1 22 2D FE C9 :42  
 :50A0=CD A8 50 CD 53 61 18 A4 :02  
 :50A8=2A 2D FE 3E C3 CD B2 5F :34  
 :50B0=2A 2B FE CD FB 5F 2A B3 :57  
 :50B8=FD 2B 2B 22 2D FE C9 FE :67  
 :50C0=3B C2 52 66 3E C3 2A 23 :03  
 :50C8=FE CD B2 5F 2A B3 FD 2B :E1  
 :50D0=2B 22 23 FE C3 53 61 FE :E3  
 :50D8=99 C2 52 66 CD 53 61 3D :D1  
 :50E0=C2 52 66 21 BD FD CD FB :1D  
 :50E8=44 B7 28 0D 3E 98 32 BD :F5  
 :50F0=FD 21 00 00 22 CA FD 18 :1F  
 :50F8=1E 3A BD FD 47 E6 0F FE :4C

:5100=08 C2 52 66 2A CA FD 78 :EB  
 :5108=87 38 0C 3E 18 CD 6E 51 :AD  
 :5110=3E C3 DC B2 5F 18 13 3E :57  
 :5118=C3 CD B2 5F 2A B3 FD 2B :A6  
 :5120=2B 22 CA FD 21 BD FD CD :BC  
 :5128=74 44 CD 53 61 FE 3B C2 :34  
 :5130=52 66 C3 53 61 FE 3B C2 :2A  
 :5138=52 66 3A 33 FE B7 28 12 :14  
 :5140=3E C3 2A 34 FE CD B2 5F :3B  
 :5148=2A B3 FD 2B 2B 22 34 FE :84  
 :5150=18 05 3E C9 CD 97 5F C3 :AA  
 :5158=53 61 FE 3B C2 52 66 3E :A5  
 :5160=C3 2A AD FD 11 03 00 19 :C4  
 :5168=CD B2 5F C3 53 61 E5 F5 :2F  
 :5170=ED 5B B3 FD 13 13 B7 ED :C2  
 :5178=52 D1 5D 7D 87 9F BC E1 :C0  
 :5180=20 05 CD A8 5F B7 C9 37 :B0  
 :5188=C9 DD E1 2A 21 FE E5 2A :DF  
 :5190=23 FE E5 21 00 00 22 23 :6C  
 :5198=FE 2A B3 FD 22 21 FE DD :F6  
 :51A0=E9 DD E1 2A 23 FE 7C B5 :23  
 :51A8=C4 FB 5F E1 22 23 FE E1 :23  
 :51B0=22 21 FE DD E9 3A DD FD :1B  
 :51B8=D6 A1 DA F3 53 FE 05 D2 :6C  
 :51C0=F3 53 87 5F 16 00 21 D1 :34  
 :51C8=51 19 5E 23 56 D5 C3 53 :2C  
 :51D0=61 DB 51 88 52 31 53 E1 :CC  
 :51D8=53 E4 53 D6 BF 28 05 FE :4A  
 :51E0=01 C2 66 66 F5 CD 53 61 :05  
 :51E8=FE F0 C2 66 66 CD 53 61 :FD  
 :51F0=D6 BF 28 0B FE 01 28 07 :F6  
 :51F8=CD 45 60 3E 02 18 05 F5 :C4  
 :5200=CD 53 61 F1 D1 5F 3A DD :B9  
 :5208=FD FE 3B 28 43 FE 2B 28 :F2  
 :5210=05 D6 2D C2 66 66 F5 D5 :60  
 :5218=E5 CD 53 61 CD F3 53 E1 :5A  
 :5220=D1 D5 CD 56 52 D1 15 16 :17  
 :5228=DD 20 02 16 FD 1E 19 F1 :3A  
 :5230=D5 B7 28 0A 3E 5F 21 16 :92  
 :5238=00 CD B2 5F 18 0E 11 5F :74  
 :5240=2F CD A8 5F 3E 16 21 FF :77  
 :5248=13 CD B2 5F D1 C3 A8 5F :8C  
 :5250=CD 56 52 C3 53 61 7B BA :21  
 :5258=C8 B7 20 0C 11 E5 DD CD :4B  
 :5260=A8 5F 11 E1 FD C3 A8 5F :C0  
 :5268=3D 20 0C 11 E5 FD CD A8 :D1  
 :5270=5F 11 E1 DD C3 A8 5F 15 :0D  
 :5278=3E DD 20 02 3E FD E5 CD :2A  
 :5280=97 5F E1 3E 21 C3 B2 5F :0A  
 :5288=16 DD D6 BF 28 06 16 FD :C9  
 :5290=3D C2 66 66 D5 CD 53 61 :21  
 :5298=FE F0 C2 66 66 CD 53 61 :FD  
 :52A0=3D C2 66 66 21 BD FD CD :73  
 :52A8=FB 44 B7 C4 B8 64 2A CA :CA  
 :52B0=FD 3A BD FD E6 0F D6 02 :BE  
 :52B8=28 04 3D C2 66 66 E5 CD :A9  
 :52C0=53 61 FE 5B 20 44 CD 53 :91  
 :52C8=61 CD 39 54 3A E2 FD 3D :11  
 :52D0=C2 81 66 3A DD FD FE 5D :18  
 :52D8=C2 66 66 CD 53 61 FD 7E :8A  
 :52E0=FD FE 02 28 1B B7 28 06 :25  
 :52E8=21 21 53 CD D3 5E E1 CD :41  
 :52F0=81 5E 21 25 53 CD D3 5E :76  
 :52F8=D1 1E E1 CD A8 5F 18 16 :D2  
 :5300=E1 FD 7E FE 85 6F 30 01 :7F  
 :5308=24 FE E1 E3 EB 1E 2A CD :E6  
 :5310=A8 5F E1 CD AA 5F 3A DD :D5  
 :5318=FD FE 3B CA 53 61 C3 66 :DD  
 :5320=66 03 3A B9 B9 0B 5F 16 :95  
 :5328=00 21 B9 B9 19 5E 23 56 :83

つづく



## リスト4-13 つづき

```

:5330=D5 3D C2 66 66 21 BD FD :7B
:5338=CD FB 44 B7 C4 B8 64 2A :CD
:5340=CA FD 3A BD FD E6 0F FE :AE
:5348=02 C2 66 66 E5 CD 53 61 :F6
:5350=FE 5B 20 5C CD 53 61 CD :23
:5358=39 54 3A E2 FD 3D C2 81 :26
:5360=66 3A DD FD FE 5D C2 66 :FD
:5368=66 FD 7E FD FE 02 28 33 :39
:5370=B7 28 06 21 21 53 CD D3 :1A
:5378=5E E1 CD 81 5E 21 D4 53 :33
:5380=CD D3 5E CD 53 61 FE F0 :6D
:5388=C2 66 66 CD 53 61 16 DD :02
:5390=FE BF 28 02 16 FD 1E E5 :FD
:5398=CD A8 5F 21 DC 53 CD D3 :C4
:53A0=5E 18 26 E1 FD 7E FE 85 :7B
:53A8=6F 30 01 24 E5 CD 53 61 :2A
:53B0=FE F0 C2 66 66 CD 53 61 :FD
:53B8=16 DD FE BF 28 02 16 FD :ED
:53C0=1E 22 CD A8 5F E1 CD AA :6C
:53C8=5F CD 53 61 FE 3B CA 53 :36
:53D0=61 C3 66 66 07 5F 16 00 :6C
:53D8=21 B9 B9 19 04 D1 73 23 :17
:53E0=72 1E 23 21 1E 2B 16 DD :10
:53E8=FE BF 28 02 16 FD CD A8 :6F
:53F0=5F 18 D6 CD 01 54 FE 3B :A8
:53F8=CA 53 61 C3 81 66 CD 75 :6A
:5400=5E CD 13 54 3A E2 FD 3D :E8
:5408=C2 81 66 3A DD FD C9 AF :35
:5410=32 E1 FD CD 39 54 FD 7E :E5
:5418=FD B7 C8 21 30 54 3D 28 :86
:5420=0C 21 37 54 FD 7E FE B7 :E8
:5428=28 03 21 34 54 C3 D3 5E :C8
:5430=FD 3A B9 B9 FE 3E B8 FF :9C
:5438=AF CD EB 54 3A DD FD FE :CD
:5440=A6 28 21 FE A7 28 07 FE :C1
:5448=21 28 03 FE 7C C0 CD E8 :3B
:5450=54 21 7A 54 CD BA 5E 30 :58
:5458=E3 FD 7E FE FD B6 01 FD :0D
:5460=77 FE 18 D8 CD E8 54 21 :8F
:5468=B1 54 CD BA 5E 30 CD FD :E4
:5470=7E FE FD AE 01 FD 77 FE :9A
:5478=18 C2 8A 54 8D 54 92 54 :7F
:5480=95 54 9A 54 A2 54 A8 54 :C9
:5488=AB 54 02 D1 B2 04 21 BB :64
:5490=BB B6 02 F6 BA 04 21 B9 :01
:5498=B9 B6 F9 3A B9 B9 21 BB :F0
:54A0=BB B6 FB 3A B9 B9 F6 BA :C8
:54A8=02 F6 B8 FB 3A BB BB F6 :51
:54B0=B8 C1 54 C4 54 C9 54 CC :CE
:54B8=54 D1 54 D9 54 DF 54 E2 :BB
:54C0=54 02 D1 AA 04 21 BB BB :6C
:54C8=AE 02 EE BA 04 21 B9 B9 :EF
:54D0=AE F9 3A B9 B9 21 BB BB :EA
:54D8=AE FB 3A B9 B9 EE BA 02 :FF
:54E0=EE B8 FB 3A BB BB EE B8 :F7
:54E8=CD 53 61 CD 48 55 3A DD :02
:54F0=FD FE A8 28 03 FE 26 C0 :B2
:54F8=CD 45 55 21 0E 55 CD BA :72
:5500=5E 30 EB FD 7E FE FD A6 :95
:5508=01 FD 77 FE 18 E0 1E 55 :DE
:5510=21 55 26 55 29 55 2E 55 :F2
:5518=36 55 3C 55 3F 55 02 D1 :83
:5520=A2 04 21 BB BB A6 02 E6 :CB
:5528=BA 04 21 B9 B9 A6 F9 3A :2A
:5530=B9 B9 21 BB BB A6 FB 3A :E4
:5538=B9 B9 E6 BA 02 E6 B8 FB :AD
:5540=3A BB BB E6 B8 CD 53 61 :CF
:5548=3A DD FD FE A9 28 08 FE :E9
:5550=5E 28 04 FE 7E 20 27 CD :1A
:5558=53 61 CD 7E 55 FD 7E FD :CC

```

```

:5560=B7 28 11 3D 28 08 FD 7E :D8
:5568=FE 2F FD 77 FE C9 21 79 :02
:5570=55 C3 D3 5E 3E 2F C3 97 :10
:5578=5F FC 3A B9 B9 2F CD FE :01
:5580=55 3A DD FD FE 3D 28 57 :23
:5588=FE F3 28 42 FE 3E 28 2D :EC
:5590=FE F4 28 1A FE 3C 28 0E :A4
:5598=FE F2 C0 CD FB 55 CD A7 :41
:55A0=5E CD 41 56 18 10 CD FB :B2
:55A8=55 CD 41 56 18 1A CD FB :B3
:55B0=55 CD 41 56 06 02 3F 9F :9F
:55B8=21 B5 55 18 32 CD FB 55 :92
:55C0=CD A7 5E CD 41 56 06 01 :3D
:55C8=9F 21 C7 55 18 21 CD FB :DD
:55D0=55 CD 41 56 06 04 28 02 :ED
:55D8=3E FF 21 D5 55 18 10 CD :7D
:55E0=FB 55 CD 41 56 06 05 28 :E7
:55E8=02 3E FF 2F 21 E6 55 47 :11
:55F0=FD 7E FD B7 CA D3 5E FD :27
:55F8=70 FE C9 CD 53 61 CD 5B :E0
:5600=57 3A DD FD FE 2B 28 4A :06
:5608=FE 2D 28 2D FE AA 28 16 :66
:5610=FE AB C0 CD 58 57 21 68 :6E
:5618=56 CD BA 5E 30 E3 21 A4 :13
:5620=56 CD DE 5E 18 DB CD 58 :77
:5628=57 21 A9 56 CD BA 5E 30 :8C
:5630=D0 21 E0 56 CD DE 5E 18 :48
:5638=C8 CD 58 57 CD 41 56 18 :C0
:5640=C0 21 E5 56 CD BA 5E D0 :D1
:5648=FD 7E FE FD 96 01 FD 77 :81
:5650=FE C9 CD 58 57 21 21 57 :DC
:5658=CD BA 5E 30 A4 FD 7E FE :32
:5660=FD 86 01 FD 77 FE 18 99 :A7
:5668=78 56 7C 56 81 56 84 56 :51
:5670=8A 56 92 56 98 56 9D 56 :A9
:5678=03 57 F1 9A 04 21 BB BB :80
:5680=9E 02 DE BA 05 57 3A B9 :87
:5688=B9 9A F9 3A B9 B9 21 BB :D4
:5690=BB 9E FB 3A B9 B9 DE BA :98
:5698=04 57 3E B8 9A FA 3E B8 :DB
:56A0=21 BB BB 9E FC 3E B8 DE :05
:56A8=BA B9 56 BC 56 C1 56 C4 :B6
:56B0=56 C9 56 D1 56 D7 56 DA :A3
:56B8=56 02 D1 8A 04 21 BB BB :4E
:56C0=8E 02 CE BA 04 21 B9 B9 :AF
:56C8=8E F9 3A B9 B9 21 BB BB :CA
:56D0=8E FB 3A B9 B9 CE BA 02 :BF
:56D8=CE BA FB 3A BB BB CE B8 :B9
:56E0=FC 3E B8 CE BA F5 56 F9 :BE
:56E8=56 FE 56 01 57 07 57 0F :6F
:56F0=57 15 57 1A 57 03 57 F1 :7F
:56F8=92 04 21 BB BB 96 02 D6 :9B
:5700=BA 05 57 3A B9 B9 92 F9 :4D
:5708=3A B9 B9 21 BB BB 96 FB :D4
:5710=3A B9 B9 D6 BA 04 57 3E :D5
:5718=B8 92 FA 3E B8 21 BB BB :D1
:5720=96 31 57 34 57 39 57 3C :75
:5728=57 41 57 49 57 4F 57 52 :87
:5730=57 02 D1 82 04 21 BB BB :47
:5738=86 02 C6 BA 04 21 B9 B9 :9F
:5740=86 F9 3A B9 B9 21 BB BB :C2
:5748=86 FB 3A B9 B9 C6 BA 02 :AF
:5750=C6 B8 FB 3A BB BB C6 B8 :A7
:5758=CD 53 61 CD 79 59 3A DD :37
:5760=FD FE 2A 28 73 FE 2F 28 :15
:5768=56 FE 25 28 39 FE F1 28 :F1
:5770=1C FE F5 C0 CD 76 59 21 :8C
:5778=F3 57 CD BA 5E 30 DF FD :3B
:5780=7E FE FD 56 01 CD BE 66 :C1
:5788=FD 77 FE 18 D1 CD 76 59 :F7

```

つづく



リスト4-13 つづき

:5790=21 41 58 CD BA 5E 30 C6 :95  
 :5798=FD 7E FE FD 56 01 CD B8 :52  
 :57A0=66 FD 77 FE 18 B8 CD 76 :EB  
 :57A8=59 21 8F 58 CD BA 5E 30 :76  
 :57B0=AD FD 7E FE FD 56 01 CD :47  
 :57B8=B2 66 FD 77 FE 18 9F CD :0E  
 :57C0=76 59 21 DD 58 CD BA 5E :0A  
 :57C8=30 94 FD 7E FE FD 56 01 :91  
 :57D0=CD AC 66 FD 77 FE 18 86 :EF  
 :57D8=CD 76 59 21 2B 59 CD BA :C8  
 :57E0=5E D2 5E 57 FD 7E FE FD :5B  
 :57E8=56 01 CD A6 66 FD 77 FE :A2  
 :57F0=C3 5E 57 03 58 09 58 10 :44  
 :57F8=58 16 58 1E 58 28 58 31 :ED  
 :5800=58 38 58 05 57 F1 CD 21 :23  
 :5808=00 06 21 BB BB CD 1E 00 :88  
 :5810=05 16 BA CD 21 00 07 57 :21  
 :5818=3A B9 B9 CD 21 00 F7 3A :CB  
 :5820=B9 B9 21 BB BB CD 1E 00 :F4  
 :5828=F8 3A B9 B9 16 BA CD 21 :62  
 :5830=00 06 57 3E B8 CD 21 00 :41  
 :5838=F8 3E B8 21 BB BB CD 1E :70  
 :5840=00 51 58 57 58 5E 58 64 :72  
 :5848=58 6C 58 76 58 7F 58 86 :47  
 :5850=58 05 57 F1 CD 1B 00 06 :93  
 :5858=21 BB BB CD 18 00 05 16 :97  
 :5860=BA CD 1B 00 07 57 3A B9 :F3  
 :5868=B9 CD 1B 00 F7 3A B9 B9 :44  
 :5870=21 BB BB CD 18 00 F8 3A :AE  
 :5878=B9 B9 16 BA CD 1B 00 06 :30  
 :5880=57 3E B8 CD 1B 00 F8 3E :6B  
 :5888=B8 21 BB BB CD 18 00 9F :D3  
 :5890=58 A5 58 AC 58 B2 58 BA :1D  
 :5898=58 C4 58 CD 58 D4 58 05 :CA  
 :58A0=57 F1 CD 15 00 06 21 BB :0C  
 :58A8=BB CD 12 00 05 16 BA CD :3C  
 :58B0=15 00 07 57 3A B9 B9 CD :EC  
 :58B8=15 00 F7 3A B9 B9 21 BB :94  
 :58C0=BB CD 12 00 F8 3A B9 B9 :3E  
 :58C8=16 BA CD 15 00 06 57 3E :4D  
 :58D0=B8 CD 15 00 F8 3E B8 21 :A9  
 :58D8=BB BB CD 12 00 ED 58 F3 :8D  
 :58E0=58 FA 58 00 59 08 59 12 :76  
 :58E8=59 1B 59 22 59 05 57 F1 :95  
 :58F0=CD 0F 00 06 21 BB BB CD :46  
 :58F8=0C 00 05 16 BA CD 0F 00 :BD  
 :5900=07 57 3A B9 B9 CD 0F 00 :E6  
 :5908=F7 3A B9 B9 21 BB BB CD :07  
 :5910=0C 00 F8 3A B9 B9 16 BA :80  
 :5918=CD 0F 00 06 57 3E B8 CD :FC  
 :5920=0F 00 F8 3E B8 21 BB BB :94  
 :5928=CD 0C 00 3B 59 40 59 47 :4D  
 :5930=59 4D 59 54 59 5E 59 67 :CA  
 :5938=59 6D 59 04 D1 CD 09 00 :CA  
 :5940=06 21 BB BB CD 06 00 05 :75  
 :5948=16 BA CD 09 00 06 21 B9 :86  
 :5950=B9 CD 06 00 F7 3A B9 B9 :2F  
 :5958=21 BB BB CD 06 00 F8 3A :9C  
 :5960=B9 B9 16 BA CD 09 00 05 :1D  
 :5968=16 B8 CD 09 00 F8 3A BB :91  
 :5970=BB 16 B8 CD 09 00 CD 53 :7F  
 :5978=61 3A DD FD FE 2B 28 2B :F1  
 :5980=FE 2D 20 2A CD AB 59 FD :43  
 :5988=7E FD B7 28 18 3D 28 09 :E0  
 :5990=FD 7E FE ED 44 FD 77 FE :1C  
 :5998=C9 21 9F 59 C3 D3 5E FB :D1  
 :59A0=3A B9 B9 ED 44 11 44 ED :1F  
 :59A8=C3 A8 5F CD 53 61 FE 28 :71  
 :59B0=20 26 CD 53 61 CD 39 54 :21  
 :59B8=FE 2C 20 14 CD 16 54 CD :62

:59C0=9A 5E CD 53 61 CD 0F 54 :A9  
 :59C8=3A DD FD 18 EB 3A DD FD :2B  
 :59D0=FE 29 C2 81 66 C3 53 61 :47  
 :59D8=FE 3F 20 53 CD 53 61 FE :2F  
 :59E0=28 C2 81 66 CD 53 61 CD :1F  
 :59E8=13 54 3A DD FD FE 3B C2 :76  
 :59F0=81 66 3E B7 CD 97 5F 23 :C2  
 :59F8=E5 3E CA 21 00 00 CD B2 :8D  
 :5A00=5F CD 53 61 CD 0F 54 CD :DD  
 :5A08=9A 5E 3A DD FD FE 2C C2 :F8  
 :5A10=81 66 3E C3 21 00 00 CD :D6  
 :5A18=B2 5F 2B 2B E3 CD FB 5F :71  
 :5A20=CD 53 61 CD 0F 54 CD 9A :18  
 :5A28=5E E1 CD FB 5F 18 9E FE :1A  
 :5A30=B9 28 05 FE 40 C2 37 5B :78  
 :5A38=CD 53 61 FE 5B C2 81 66 :83  
 :5A40=CD 53 61 FE BF 28 4B FE :AF  
 :5A48=C0 28 4A CD 13 54 CD 9A :CD  
 :5A50=5E 3A DD FD FE 2C C2 81 :DF  
 :5A58=66 CD 53 61 3E F5 CD 97 :7E  
 :5A60=5F CD 0F 54 3A DD FD FE :A1  
 :5A68=5D C2 81 66 11 6F E1 CD :34  
 :5A70=A8 5F CD 53 61 FE F0 20 :96  
 :5A78=14 CD 9A 5E 3E E5 CD 97 :60  
 :5A80=5F CD 53 61 CD 0F 54 11 :21  
 :5A88=77 E1 C3 A8 5F 3E 7E C3 :A1  
 :5A90=97 5F 3E DD 21 3E FD F5 :62  
 :5A98=CD 53 61 FE 2B 28 16 FE :E6  
 :5AA0=2D 28 0F 2E 00 FE 2C 28 :E4  
 :5AA8=1E 26 00 FE 5D 28 4C C3 :D6  
 :5AB0=81 66 2E 2B 11 2E 23 E5 :87  
 :5AB8=CD 53 61 E1 26 00 FE 5D :E3  
 :5AC0=28 39 FE 2C C2 81 66 E5 :19  
 :5AC8=CD 53 61 FE 2B 28 05 FE :D5  
 :5AD0=2D 20 08 FE AF F5 CD 53 :17  
 :5AD8=61 F1 0E AF F5 CD 31 60 :62  
 :5AE0=F1 28 04 AF 95 6F 9F CB :3A  
 :5AE8=7D 28 01 2F B7 C4 DD 65 :92  
 :5AF0=D1 55 EB 3A DD FD FE 5D :80  
 :5AF8=C2 81 66 E5 CD 53 61 FE :0D  
 :5B00=F0 20 19 CD 53 61 CD 13 :8A  
 :5B08=54 E1 D1 7A D5 E5 2E 77 :DF  
 :5B10=CD B2 5F E1 D1 7D B7 C8 :8C  
 :5B18=5D C3 A8 5F CD 88 5F E1 :BC  
 :5B20=D1 7A D5 E5 2E 7E CD B2 :30  
 :5B28=5F E1 D1 7D B7 28 05 5D :CF  
 :5B30=CD A8 5F AF C3 81 5E FE :23  
 :5B38=BA 28 04 FE F6 20 3F CD :06  
 :5B40=53 61 FE 5B C2 81 66 CD :83  
 :5B48=53 61 CD 13 54 3A DD FD :FC  
 :5B50=FE 5D C2 81 66 CD 53 61 :85  
 :5B58=FE F0 20 1A CD 9A 5E 3E :2B  
 :5B60=F5 CD 97 5F CD 53 61 CD :06  
 :5B68=0F 54 11 4A D1 CD A8 5F :63  
 :5B70=11 79 ED C3 A8 5F 3E 4F :CE  
 :5B78=21 ED 78 C3 B2 5F FE AC :04  
 :5B80=28 07 FE AD 20 48 3E 01 :81  
 :5B88=FE AF F5 CD 53 61 FE 28 :49  
 :5B90=C2 81 66 CD 53 61 3D C2 :29  
 :5B98=81 66 21 BD FD CD FB 44 :CE  
 :5BA0=B7 C4 B8 64 3A BD FD E6 :71  
 :5BA8=0F FE 02 C2 81 66 2A CA :AC  
 :5BB0=FD 3E 21 CD B2 5F F1 C6 :F1  
 :5BB8=34 CD 97 5F AF CD 81 5E :52  
 :5BC0=21 CC 5B CD D3 5E CD 53 :66  
 :5BC8=61 C3 CD 59 FF 7E FE AE :73  
 :5BD0=DA 62 5C FE B4 30 2E D6 :7E  
 :5BD8=AE F5 CD 53 61 FE 28 C2 :0C  
 :5BE0=81 66 CD 53 61 CD B5 59 :43  
 :5BE8=CD 16 54 F1 5F 16 00 21 :BE

つづく



## リスト4-13 つづき

```

:5BF0=FF 5B 19 7E FE CB C2 97 :13
:5BF8=5F 57 1E 2F C3 A8 5F 17 :E4
:5C00=07 1F 0F CB 27 FE B9 30 :0E
:5C08=59 D6 B4 F5 CD 53 61 FE :57
:5C10=28 C2 81 66 CD 53 61 FE :50
:5C18=29 28 08 CD B5 59 CD 16 :17
:5C20=54 18 0A CD 53 61 CD 88 :4C
:5C28=5F AF CD 81 5E F1 20 04 :CF
:5C30=3E 9F 18 13 F5 11 00 3E :4C
:5C38=CD A8 5F F1 3D 20 0B 11 :3E
:5C40=01 20 CD A8 5F 3E 2F C3 :25
:5C48=97 5F 21 00 00 3D 3E F2 :84
:5C50=28 02 3E E2 CD B2 5F 3E :66
:5C58=2F CD 97 5F 2B 2B 2B C3 :36
:5C60=FB 5F 2A B9 FD B7 CA 39 :F4
:5C68=5D 3D C2 34 5D 21 BD FD :C8
:5C70=CD FB 44 B7 20 19 2A CA :F0
:5C78=FD 3A BD FD E6 0F FE 01 :E5
:5C80=CA 39 5D FE 08 CA 81 66 :17
:5C88=FE 09 CA 71 5D 18 13 2A :F4
:5C90=A7 FD CD 02 64 FE 28 CA :C7
:5C98=48 5D CD B8 64 3E 02 21 :EF
:5CA0=00 00 F5 E5 CD 53 61 FE :59
:5CA8=5B 20 62 CD 53 61 CD 39 :64
:5CB0=54 3A DD FD FE 5D C2 81 :06
:5CB8=66 FD 7E FD FE 02 28 3D :43
:5CC0=B7 28 06 21 30 54 CD D3 :2A
:5CC8=5E E1 CD 81 5E 21 2C 5D :95
:5CD0=CD D3 5E CD 9A 5E CD 53 :E3
:5CD8=61 FE F0 20 1A F1 FE 03 :7B
:5CE0=CA 81 66 CD 9A 5E 3E E5 :99
:5CE8=CD 97 5F CD 53 61 CD 0F :20
:5CF0=54 11 77 E1 C3 A8 5F F1 :78
:5CF8=3E 7E C3 97 5F E1 FD 7E :D1
:5D00=FE 85 6F 30 01 24 E5 CD :F9
:5D08=9A 5E CD 53 61 FE F0 20 :87
:5D10=14 E1 F1 FE 03 CA 81 66 :98
:5D18=E5 CD 53 61 CD 13 54 E1 :7B
:5D20=3E 32 C3 B2 5F E1 F1 3E :54
:5D28=01 C3 81 5E 07 5F 16 00 :1F
:5D30=21 B9 B9 19 CD 31 60 18 :22
:5D38=0A 7C B7 C4 EF 65 E5 CD :07
:5D40=53 61 E1 3E 02 C3 81 5E :77
:5D48=21 00 00 22 1F FE 01 0C :6D
:5D50=00 21 BE FD CD 37 61 CD :0E
:5D58=7A 5D 01 0C 00 21 BE FD :C0
:5D60=CD 47 61 2A 1F FE 22 CA :A8
:5D68=FD 21 BD FD 36 89 C3 74 :CE
:5D70=44 3A BD FD 07 38 D4 22 :6D
:5D78=1F FE CD 53 61 FE 28 C2 :86
:5D80=81 66 CD 88 5F 3A E0 FD :B2
:5D88=B7 F5 3E C5 C4 97 5F 2A :93
:5D90=1F FE E5 CD 53 61 FE 3B :BC
:5D98=28 4D FE 29 CA 4C 5E 11 :21
:5DA0=01 36 D5 18 04 D5 CD 53 :1D
:5DA8=61 CD 0F 54 CD 9A 5E 3E :94
:5DB0=F5 CD 97 5F D1 3A DD FD :9D
:5DB8=FE 3B 28 16 FE 29 28 12 :D8
:5DC0=FE 2C C2 81 66 1C 7B FE :68
:5DC8=21 38 DA CC A3 65 1E 21 :46
:5DD0=18 D3 F5 D5 3E 21 2A AF :ED
:5DD8=FD CD B2 5F D1 CD A8 5F :80
:5DE0=F1 FE 29 28 74 18 0D 3E :17
:5DE8=AF CD 97 5F 3E 32 2A AF :BB
:5DF0=FD CD B2 5F CD 53 61 FE :5A
:5DF8=2C 28 28 FE BF 20 10 11 :7A
:5E00=E5 DD CD A8 5F 3E E1 CD :82
:5E08=97 5F CD 53 61 18 00 CD :64
:5E10=0B 60 3E 21 CD B2 5F 3A :E2
:5E18=DD FD FE 29 28 3B FE 2C :8E

```

```

:5E20=C2 81 66 CD 53 61 FE C0 :E8
:5E28=20 10 11 E5 FD CD A8 5F :F7
:5E30=3E D1 CD 97 5F CD 53 61 :53
:5E38=18 08 CD 0B 60 3E 11 CD :74
:5E40=B2 5F 3A DD FD FE 29 C2 :0E
:5E48=81 66 18 0D 3E AF CD 97 :5D
:5E50=5F 3E 32 2A AF FD CD B2 :24
:5E58=5F D1 2A B3 FD 23 22 1F :6E
:5E60=FE EB 3E CD CD B2 5F F1 :C3
:5E68=3E C1 C4 97 5F 3E 00 CD :C4
:5E70=81 5E C3 53 61 FD 21 E3 :57
:5E78=FD AF 32 E2 FD 32 E1 FD :CD
:5E80=C9 FD 77 00 FD 75 01 FD :AD
:5E88=74 02 11 03 00 FD 19 21 :C1
:5E90=E2 FD 34 7E FE 14 D2 81 :F6
:5E98=66 C9 FD 2B FD 2B FD 2B :A7
:5EA0=E5 21 E2 FD 35 E1 C9 FD :C1
:5EA8=E5 D1 21 FD FF 19 06 03 :F5
:5EB0=2B 1B 1A 4E EB 71 12 10 :2C
:5EB8=F7 C9 FD 7E FA 5F 87 83 :9E
:5EC0=FD 86 FD FE 08 CA 6F 5F :1E
:5EC8=87 5F 16 00 19 5E 23 56 :EC
:5ED0=EB 18 0B 11 03 00 FD 19 :38
:5ED8=EB 21 E2 FD 34 EB 7E 47 :CF
:5EE0=B7 FC 74 5F C5 E5 CD 61 :5E
:5EE8=44 E1 C1 23 C5 7E FE B8 :02
:5EF0=20 05 FD 7E FB 18 62 FE :13
:5EF8=B9 20 0C FD 7E FB E5 CD :0D
:5F00=61 44 FD 7E FC 18 17 FE :49
:5F08=BA 20 05 FD 7E FE 18 49 :B9
:5F10=FE BB 20 1B FD 7E FE E5 :52
:5F18=CD 61 44 FD 7E FF CD 61 :1A
:5F20=44 2A B3 FD 23 23 22 B3 :39
:5F28=FD E1 23 C1 05 18 30 FE :0D
:5F30=CD 20 26 E5 CD 61 44 E1 :4B
:5F38=23 5E 23 56 E5 2A AD FD :B3
:5F40=19 E5 7D CD 61 44 F1 CD :AB
:5F48=61 44 2A B3 FD 23 23 23 :E8
:5F50=22 B3 FD E1 C1 05 05 18 :96
:5F58=06 E5 CD 9D 5F E1 C1 10 :66
:5F60=8A FD 36 FA 00 3E FF 32 :26
:5F68=E1 FD CD 9A 5E B7 C9 CD :F0
:5F70=9A 5E 37 C9 ED 44 47 3A :AA
:5F78=E1 FD B7 78 C8 3C C5 E5 :BB
:5F80=CD 9D 5F E1 C1 3E F5 C9 :67
:5F88=3A E1 FD B7 3E F5 C2 97 :5B
:5F90=5F 3E FF 32 E1 FD C9 F5 :6A
:5F98=AF CD 61 44 F1 CD 61 44 :84
:5FA0=2A B3 FD 23 22 B3 FD C9 :98
:5FA8=63 6A E5 3E 02 CD 61 44 :64
:5FB0=18 0B E5 F5 3E 03 CD 61 :6C
:5FB8=44 F1 CD 9D 5F E1 7D E5 :41
:5FC0=CD 61 44 F1 CD 61 44 2A :FF
:5FC8=B3 FD 23 23 22 B3 FD C9 :91
:5FD0=21 BE FD 11 CD FD 06 0C :C9
:5FD8=7E 12 B7 C8 23 13 10 F8 :4D
:5FE0=78 12 C9 CD 61 44 21 CD :B3
:5FE8=FD 7E B7 CA 61 44 E5 CD :53
:5FF0=61 44 E1 23 18 F3 E5 3E :D7
:5FF8=80 18 03 E5 3E 81 CD 61 :6D
:6000=44 E1 7D E5 CD 61 44 F1 :EA
:6008=C3 61 44 CD 45 60 3A DD :F1
:6010=FD FE 2B 28 11 FE 2D C0 :4A
:6018=E5 CD 53 61 CD 45 60 EB :C3
:6020=E1 B7 ED 52 18 E8 E5 CD :89
:6028=53 61 CD 45 60 D1 19 18 :28
:6030=DD 21 DF FD 7E F5 36 00 :83
:6038=CD 45 60 F1 32 DF FD 7C :ED
:6040=B7 C2 EF 65 C9 21 DF FD :93
:6048=46 3A DD FD FE BB 28 07 :42

```

つづく



リスト4-13 つづき

```

:6050=FE BC 28 0F C3 73 60 C5 :4C
:6058=36 00 CD 53 61 CD 73 60 :57
:6060=6C 18 09 C5 36 00 CD 53 :A8
:6068=61 CD 73 60 F1 32 DF FD :00
:6070=26 00 C9 3A DD FD B7 20 :DA
:6078=05 2A B9 FD 18 43 3D 20 :9D
:6080=15 21 BD FD CD FB 44 B7 :B3
:6088=C4 CA 64 3A BD FD E6 0F :DB
:6090=3D C2 22 66 18 28 FE 2D :F2
:6098=C2 22 66 CD 53 61 3D C2 :CA
:60A0=36 66 21 BD FD CD FB 44 :83
:60A8=B7 20 49 3A BD FD E6 0F :09
:60B0=3D CA 36 66 3D 28 07 3D :4C
:60B8=28 04 D6 05 30 09 2A CA :34
:60C0=FD E5 CD 53 61 E1 C9 3A :47
:60C8=DF FD B7 CA 36 66 ED 5B :41
:60D0=CA FD 3A BD FD 4F E6 0F :FF
:60D8=FE 09 38 0C CD 1A 61 D2 :65
:60E0=36 66 CB 79 20 24 18 D6 :12
:60E8=CD 1A 61 DA 36 66 CB 79 :02
:60F0=20 18 18 CA 3A DF FD B7 :E7
:60F8=CA 36 66 11 00 00 CD 1A :5E
:6100=61 3E 98 30 02 3E 89 32 :62
:6108=BD FD 2A B3 FD 22 CA FD :7D
:6110=D5 21 BD FD CD 74 44 E1 :16
:6118=18 A7 2A A7 FD CD 02 64 :C0
:6120=28 13 FE 28 20 0F CD 0C :69
:6128=64 28 0A FE 29 20 06 23 :06
:6130=22 A7 FD 37 C9 B7 C9 DD :23
:6138=E1 EB 21 00 00 39 B7 ED :CA
:6140=42 F9 EB ED B0 DD E9 DD :66
:6148=E1 EB 21 00 00 39 ED B0 :C3
:6150=F9 DD E9 3A DD FD FE FF :D0
:6158=C8 2A A7 FD CD 02 64 20 :E9
:6160=22 CD 29 43 30 0A CD AE :10
:6168=42 38 F1 3E FF C3 8F 62 :5C
:6170=ED 53 A9 FD E5 EB 3A AC :9C
:6178=FD B7 C4 23 44 E1 CD 02 :8F
:6180=64 28 DE FE 2F 20 10 23 :EA
:6188=7E 2B FE 2A 7E 20 08 23 :9A
:6190=CD D6 63 38 D1 18 C5 22 :0E
:6198=BB FD EB CD 0F 64 38 05 :20
:61A0=CD 36 64 18 3D FE 24 20 :FE
:61A8=11 13 1A CD 16 64 38 05 :C2
:61B0=CD 6E 64 18 2D 2A B3 FD :BE
:61B8=18 28 FE 27 20 2B 26 00 :D6
:61C0=13 1A B7 28 0A 6F 13 1A :B2
:61C8=FE 27 20 0E 13 18 13 21 :B2
:61D0=27 00 22 CD FD 21 20 00 :54
:61D8=18 08 22 CE FD 3E 27 32 :A4
:61E0=CD FD AF 22 B9 FD C3 88 :9C
:61E8=62 FE 5F 28 05 CD 28 64 :45
:61F0=38 51 21 BE FD 06 0C 77 :EE
:61F8=23 13 1A CD 0F 64 30 09 :C9
:6200=CD 28 64 30 04 FE 5F 20 :0A
:6208=06 10 EC 06 01 18 EA 36 :41
:6210=00 ED 53 A7 FD 11 9C 62 :F3
:6218=21 BE FD 1A B7 FA 2E 62 :37
:6220=28 1D 4F 7E CD F9 63 B9 :F4
:6228=20 0C 13 23 18 ED 7E B7 :9C
:6230=20 0A 1A C3 8F 62 13 1A :25
:6238=B7 F2 36 62 13 18 D9 3E :83
:6240=01 18 4C 21 BB 63 01 1B :C0
:6248=00 ED B1 20 46 4F 13 ED :53
:6250=53 A7 FD FE 3A 20 07 1A :70
:6258=FE 3D 3E F0 18 27 FE 3C :E2
:6260=20 0B 1A D6 3C FE 03 30 :88
:6268=25 C6 F1 18 1A FE 3E 20 :6A
:6270=0B 1A D6 3D FE 02 30 16 :7E
:6278=C6 F4 18 0B FE 40 20 0E :49

```

```

:6280=1A FE 40 3E F6 20 07 13 :C6
:6288=ED 53 A7 FD 18 01 79 32 :A8
:6290=DD FD C9 6F 26 00 22 CD :27
:6298=FD C3 BD 65 41 4E 44 A8 :5D
:62A0=41 54 BE 42 52 45 41 4B :B8
:62A8=8B 42 59 9A 42 59 54 45 :F4
:62B0=91 43 41 52 52 59 B4 43 :09
:62B8=4F 4E 53 8C 44 41 54 41 :96
:62C0=8E 44 45 43 AD 44 45 43 :D3
:62C8=4A B3 44 45 58 A5 44 45 :0C
:62D0=42 55 47 84 45 4C 53 45 :8B
:62D8=95 45 4C 53 45 49 46 BD :0A
:62E0=45 58 49 54 9E 46 4F 52 :BF
:62E8=98 47 4F 9C 47 4F 54 4F :03
:62F0=9D 48 49 BB 49 46 93 49 :54
:62F8=4E 43 AC 49 4E 43 4C 55 :B8
:6300=44 45 81 49 4E 4C 49 4E :84
:6308=45 90 49 4E 58 A4 49 58 :09
:6310=BF 49 59 C0 4C 44 58 A2 :AB
:6318=4C 4F 4F 50 9B 4C 4F 57 :C7
:6320=BC 4D 45 4D 4F 52 59 B9 :4E
:6328=4D 49 4E 55 53 AB 4E 4F :D4
:6330=54 A9 4F 52 A7 4F 56 45 :2F
:6338=52 46 4C 4F 57 B8 50 41 :D3
:6340=52 49 54 59 B7 50 4C 55 :F0
:6348=53 AA 50 4F 52 54 BA 50 :4C
:6350=52 4F 47 80 52 45 43 55 :97
:6358=52 53 49 56 45 8F 52 45 :AF
:6360=54 55 52 4E 9F 52 4C AE :34
:6368=52 4C 43 AF 52 52 B0 52 :36
:6370=52 43 B1 53 45 54 A1 53 :26
:6378=49 47 4E B6 53 52 41 B2 :2C
:6380=53 54 4F 50 A0 53 54 58 :E5
:6388=A3 54 48 45 4E 94 54 4F :09
:6390=99 54 52 4F 46 46 86 54 :F4
:6398=52 4F 4E 85 55 4E 54 49 :B4
:63A0=4C 97 56 41 52 8D 57 48 :F8
:63A8=49 4C 45 96 57 4F 52 44 :AC
:63B0=92 58 4F 52 A6 5A 45 52 :22
:63B8=4F B5 00 21 22 23 25 26 :B5
:63C0=28 29 2A 2B 2C 2D 2E 2F :5C
:63C8=3A 3B 3C 3D 3E 3F 40 5B :06
:63D0=5D 5E 7B 7C 7D 7E 23 7E :4E
:63D8=B7 28 14 FE 2A 20 F7 23 :55
:63E0=7E B7 28 0B FE 2A 28 EF :A7
:63E8=FE 2F 20 EA B7 23 C9 CD :A7
:63F0=29 43 D8 ED 53 A9 FD 18 :42
:63F8=DE FE 61 D8 FE 7B D0 D6 :34
:6400=20 C9 7E B7 C8 FE 09 28 :15
:6408=03 FE 20 C0 23 18 F3 FE :0D
:6410=30 D8 FE 3A 3F C9 CD 0F :24
:6418=64 D0 FE 41 D8 FE 47 3F :CF
:6420=D0 FE 61 D8 FE 67 3F C9 :74
:6428=FE 41 D8 FE 5B 3F D0 FE :7D
:6430=61 D8 FE 7B 3F C9 21 00 :DB
:6438=00 1A D6 30 FE 0A D0 44 :3C
:6440=4D 29 38 12 29 38 0F 09 :39
:6448=38 0C 29 38 09 4F 06 00 :03
:6450=09 38 03 13 18 E3 2A BB :37
:6458=FD 11 CD FD 06 0F 7E CD :38
:6460=0F 64 38 49 05 04 28 03 :28
:6468=12 13 05 23 18 F0 21 00 :76
:6470=00 1A CD 16 64 D8 CD F9 :FF
:6478=63 D6 30 FE 0A 38 02 D6 :81
:6480=07 4F 06 00 7C E6 F0 20 :CE
:6488=08 29 29 29 29 09 13 18 :E0
:6490=E0 2A BB FD 11 CD FD 06 :A3
:6498=0E 7E 23 12 13 7E CD 16 :35
:64A0=64 38 0A 05 04 28 03 12 :EC
:64A8=13 05 23 18 F0 E5 AF 12 :E9

```

つづく



## リスト4-13 つづき

```

:64B0=CD FF 65 D1 21 00 00 C9 :EC
:64B8=11 C1 64 CD DC 64 36 02 :7B
:64C0=C9 76 61 72 69 61 62 6C :AA
:64C8=65 00 11 D3 64 CD DC 64 :BA
:64D0=36 01 C9 63 6F 6E 73 74 :27
:64D8=61 6E 74 00 21 00 00 22 :86
:64E0=CA FD 21 EC 64 CD BD 40 :02
:64E8=21 BD FD C9 23 4D 69 73 :F0
:64F0=73 69 6E 67 20 5C 20 6E :BB
:64F8=61 6D 65 20 3A 20 40 BE :AB
:6500=FD 00 21 08 65 C3 97 66 :4B
:6508=23 42 61 64 20 6F 70 74 :9D
:6510=69 6F 6E 20 73 77 69 74 :2D
:6518=63 68 00 21 21 65 C3 97 :CC
:6520=66 23 49 6C 6C 65 67 61 :D7
:6528=6C 20 66 75 6E 63 74 69 :15
:6530=6F 6E 20 6E 61 6D 65 00 :9E
:6538=21 41 65 11 BE FD C3 BD :13
:6540=40 23 40 22 65 20 3A 20 :A4
:6548=5C 00 21 50 65 C3 97 66 :F2
:6550=23 49 6C 6C 65 67 61 6C :DD
:6558=20 6E 61 6D 65 00 21 67 :49
:6560=65 11 BE FD C3 BD 40 23 :14
:6568=40 51 65 20 3A 20 5C 00 :CC
:6570=11 BE FD 21 79 65 C3 BD :4B
:6578=40 23 49 6C 6C 65 67 61 :B1
:6580=6C 20 6C 61 62 65 6C 20 :AC
:6588=3A 20 5C 00 21 92 65 C3 :91
:6590=BD 40 23 42 61 64 20 73 :BA
:6598=74 72 69 6E 67 20 64 61 :09
:65A0=74 61 00 21 A9 65 C3 BD :84
:65A8=40 23 54 6F 6F 20 6D 61 :83
:65B0=6E 79 20 61 72 67 75 6D :23
:65B8=65 6E 74 73 00 21 C6 65 :06
:65C0=11 CD FD C3 97 66 23 49 :07
:65C8=6C 6C 65 67 61 6C 20 63 :F4
:65D0=68 61 72 61 63 74 65 72 :4A
:65D8=20 3A 20 5C 00 11 E2 65 :2E
:65E0=18 20 64 69 73 70 6C 61 :B5
:65E8=63 65 6D 65 6E 74 00 11 :8D
:65F0=F4 65 18 0E 6F 76 65 72 :3B
:65F8=20 72 61 6E 67 65 00 11 :3E
:6600=CD FD 21 0C 66 CD BD 40 :27
:6608=21 00 00 C9 23 49 6C 6C :2E
:6610=65 67 61 6C 20 63 6F 6E :F9
:6618=73 74 61 6E 74 20 3A 20 :A4
:6620=5C 00 21 28 66 C3 97 66 :CB
:6628=23 42 61 64 20 63 6F 6E :8A
:6630=73 74 61 6E 74 00 21 3C :87
:6638=66 C3 97 66 23 42 61 64 :50

```

```

:6640=20 61 64 64 72 65 73 73 :06
:6648=20 63 6F 6E 73 74 61 6E :16
:6650=74 00 21 58 66 C3 97 66 :13
:6658=23 53 79 6E 74 61 78 20 :CA
:6660=65 72 72 6F 72 00 21 6C :B7
:6668=66 C3 97 66 23 42 61 64 :50
:6670=20 69 6E 64 65 78 20 6F :C7
:6678=70 65 72 61 74 69 6F 6E :62
:6680=00 21 87 66 C3 97 66 23 :F1
:6688=42 61 64 20 65 78 70 72 :E6
:6690=65 73 73 69 6F 6E 00 CD :5E
:6698=BD 40 C3 AC 40 C3 94 67 :6A
:66A0=C3 7C 67 C3 C4 66 C3 C6 :1C
:66A8=66 C3 D6 66 C3 DB 66 C3 :2C
:66B0=E0 66 C3 E1 66 C3 F1 66 :6A
:66B8=C3 F2 66 C3 02 67 C3 03 :0D
:66C0=67 C3 14 67 5E FE 5A 16 :71
:66C8=00 6A 67 3E 08 29 30 01 :71
:66D0=19 3D 20 F9 7D C9 CD E0 :62
:66D8=66 7B C9 CD E1 66 7B C9 :02
:66E0=56 5F AF 2E 08 CB 23 17 :9F
:66E8=BA 38 02 92 1C 2D 20 F5 :E4
:66F0=C9 56 14 15 C8 5F 7A FE :E7
:66F8=09 3E 00 D0 7B 87 15 20 :4E
:6700=FC C9 56 14 15 C8 5F 7A :E5
:6708=FE 09 3E 00 D0 7B CB 3F :9A
:6710=15 20 FB C9 22 5C FE ED :62
:6718=53 5A FE E1 22 5E FE E1 :EB
:6720=22 60 FE 3A 36 FE B7 28 :CD
:6728=0D 21 36 FE 5F 16 00 19 :F0
:6730=47 F1 77 2B 10 FB 2A 60 :6F
:6738=FE E5 2A 5A FE 7C B5 28 :BE
:6740=1E EB 21 00 00 ED 52 39 :A2
:6748=F9 C5 42 4B EB 2A 5C FE :BA
:6750=ED B0 C1 2A 5C FE E5 2A :F1
:6758=5A FE E5 21 72 67 E5 79 :95
:6760=B7 28 0B 2A 5C FE EB 21 :7A
:6768=37 FE 06 00 ED B0 2A 5E :60
:6770=FE E9 C1 D1 21 00 00 39 :D3
:6778=ED B0 F9 C9 3A 57 FE B7 :A5
:6780=20 03 76 18 FD 3D 20 05 :10
:6788=ED 7B 58 FE C9 3D 20 F2 :D6
:6790=2A 58 FE E9 9E 66 A1 66 :74
:6798=A4 66 A7 66 AA 66 AD 66 :3A
:67A0=B0 66 B3 66 B6 66 B9 66 :6A
:67A8=BC 66 BF 66 C2 66 D7 66 :AC
:67B0=DC 66 5C 67 00 00 15 67 :81
:67B8=19 67 1D 67 21 67 24 67 :17
:67C0=2A 67 37 67 3B 67 4E 67 :86
:67C8=54 67 58 67 64 67 68 67 :14
:67D0=6F 67 7D 67 8A 67 91 67 :A3
:67D8=00 00 00 00 00 00 00 :00

```









# APPENDIX





## ■ テキストのコンバート ■

『X1-DUAD』はディスク・ベースのアセンブラなので、『マシン語プログラミング入門』に掲載されているエディタ・アセンブラに比べて開発効率が向上します。

そこで、このアセンブラから『X1-DUAD』へテキストをコンバートする方法ですが、これは特に難しいところはありません。というのは、エディタ・アセンブラはテキストを単純にバイナリ形式でセーブしているだけであり、『X1-DUAD』はBASICと同じファイル構造であるからです。そのため、テキストはモニタのLコマンドかLOADM命令で読み込み、あとはシーケンシャル・ファイルとしてディスクに書き込めばよいことになります。また、『X1-DUAD』は、テキストを中間コード形式で持っていますが、アスキー形式のロード、セーブも可能です。コンバートは以下の手順で行ってください。

- ①リストA-1をアセンブルして、オブジェクトをディスクにファイル名“text loader”としてセーブ。
- ②コンバートしたいテキストが入っているテープをセットして、リストA-2のプログラムを実行する。
- ③『X1-DUAD』を起動して、LOADA命令でロードする。



## リストA-1

```

1      ;
2      ; ---- 'text loader' ----
3      ;
4 0041 load1 equ 0041H
5 0044 load2 equ 0044H
6 1321 fmprhl equ 1321H
7 0DEC cmtcom equ 0DECH
8 142F printp equ 142FH
9 04A3 cr2 equ 04A3H
10     ;
11 1472 filout equ 1472H
12 C000 startm equ 0C000H
13 C012 len1 equ startm+12H
14 3F01 free equ 0FF00H-0C000H+1
15     ;
16 BF00 ; org 0BF00H
17     ;
18 BF00 AF start: xor a
19 BF01 327214 ld (filout),a ; output device is display
20     ;
21 BF04 2100C0 ld hl,startm
22 BF07 012000 ld bc,20H
23 BF0A CD4100 call load1 ; load FCB
24 BF0D 3822 jr c,error1 ; load ok ?
25 BF0F 114FBF ld de,lmes
26 BF12 CD2113 call fmprhl ; print file name
27 BF15 ED4B12C0 ld bc,(len1) ; bc := data length
28 BF19 EB ex de,hl ; save hl to de
29 BF1A 21013F ld hl,free
30 BF1D B7 or a
31 BF1E ED42 sbc hl,bc ; if hl < bc then ' too long
32 BF20 3814 jr c,error2
33 BF22 ED4390BF ld (length),bc
34     ;
35 BF26 EB ex de,hl ; load hl from de
36 BF27 CD4400 call load2
37 BF2A 380F jr c,error3 ; load body
38     ;
39 BF2C 1157BF ld de,okmes
40 BF2F 1812 jr print2
41     ;
42 BF31 115FBF error1: ld de,emes1
43 BF34 1808 jr print
44 BF36 116EBF error2: ld de,emes2
45 BF39 1803 jr print
46 BF3B 1177BF error3: ld de,emes3
47     ;
48 BF3E 3EFF print: ld a,0FFH
49 BF40 3292BF ld (status),a
50 BF43 CDA304 print2: call cr2
51 BF46 CD2F14 call printp
52 BF49 3E01 ld a,l ; cassett stop cmd
53 BF4B CDEC0D call cmtcom
54 BF4E C9 ret
55     ;
56 BF4F 6C6F6164 lmes: defm 'loading'
57 BF53 696E67
58 BF56 00 defb 0
59 BF57 6C6F6164 okmes: defm 'load ok'
60 BF5B 206F6B
61 BF5E 00 defb 0
62 BF5F 46434220 emes1: defm 'FCB load error'
63 BF63 6C6F6164
64 BF67 20657272
65 BF6B 6F72
66 BF6D 00 defb 0
67 BF6E 746F6F20 emes2: defm 'too long'
68 BF72 6C6F6E67
69 BF76 00 defb 0
70 BF77 626F6479 emes3: defm 'body load error'
71 BF7B 206C6F61
72 BF7F 64206572
73 BF83 726F72
74 BF86 00 defb 0
75     ;
76 BF90 org 0BF90H
77 BF90 0000 length: defw 0
78 BF92 00 status: defb 0

```



リストA-2

```

1000 '
1010 ' --- convert text from editor assembler to DUAD-X1 ---
1020 '
1030 CLEAR &HBEFF
1040 LOADM"text loader"
1050 DEFUSR=&HBF00
1060 a=USR(0) : PRINT
1070 IF PEEK(&HBF92)<>0 THEN END
1080 lgth=&HBF90 : st=&HC000 : ln=1
1090 lgth=PEEK(lgth)+PEEK(lgth+1)*256
1100 PRINT "-- "; HEX$(lgth);" --"
1110 :
1120 INPUT "file name ->";na$
1130 OPEN "O",#1,na$
1140 PRINT#1,STR$(ln);
1150 WHILE lgth >= 0
1160     ch=PEEK(st)
1170     PRINT#1,CHR$(ch) ;
1180     IF ch=&HD THEN ln=ln+1 : PRINT#1,STR$(ln);"    ";
1190     st=st+1 : lgth=lgth-1
1200 WEND
1210 CLOSE

```



# ハイテックファイル

昭和60年 8 月20日 初版発行

定価 2,200円

著 者 渡辺英行，高橋秀樹  
発行人 塚本慶一郎  
発行所 株式会社エム・アイ・エー  
〒150 東京都渋谷区渋谷2-9-1 青山田中ビル  
電 話 (03)486-4500  
編集制作 アスキー出版局第二書籍編集部  
電 話 (03)486-4512

印刷・製本 東京音楽図書株式会社

ISBN4-87170-044-5 C3055 ¥2,200E



## X1マシン語 プログラミング入門

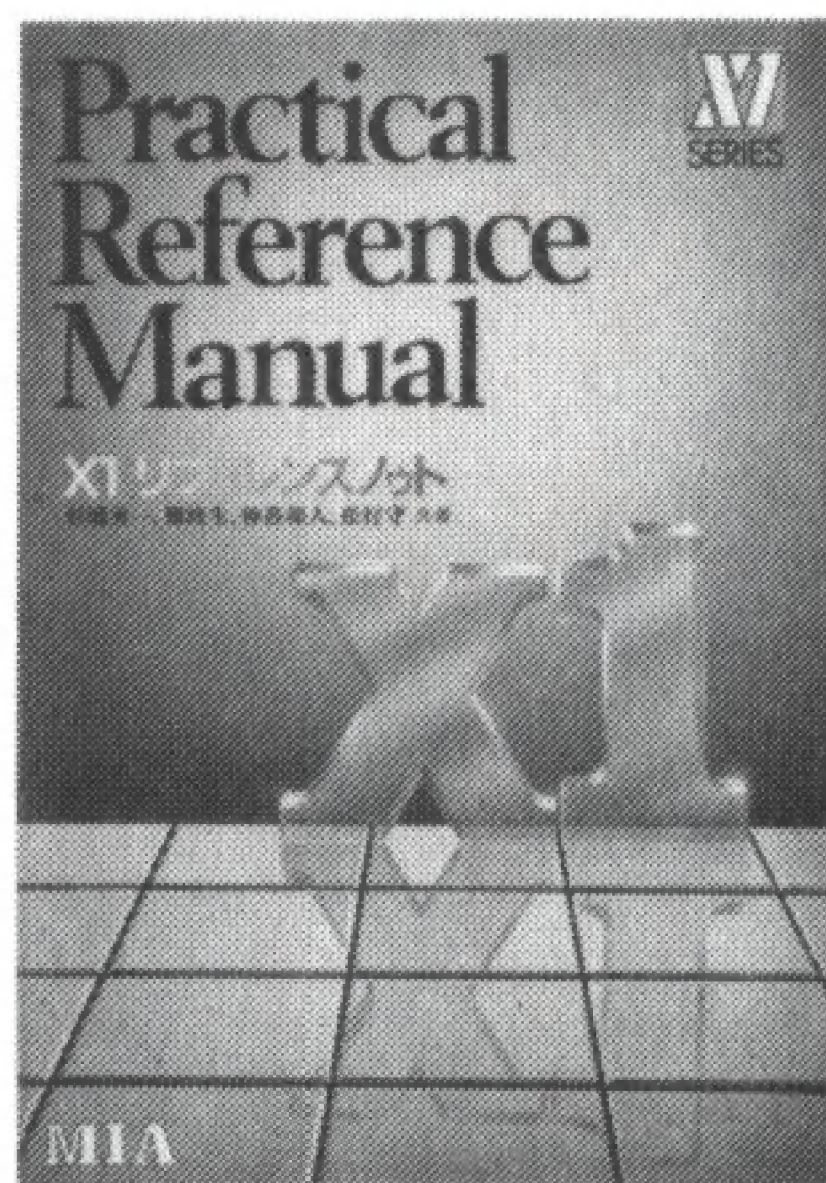
X1シリーズでマシン語を学ぶための入門書。基礎から始め、X1に特有な機能の活用までを詳しく解説している。予備知識に習熟するよりも実践的に使いこなすことを目標にしているため、一般に難しいといわれるマシン語を短期間でマスターできる。



A5判 定価 2,200円

## X1リファレンスノウト

turboを含むX1シリーズのすべての機能を解説し、活用ノウハウを紹介したテクニカル資料集。一歩進んでX1を実践的に活用したいというユーザーは必携の書。



A5判 定価 2,500円

## X1ソフトプレス① 占っちゃうから!

由緒正しき占いに、パソコンカルチャーのスパイスたっぷり。天中殺、易、おみくじなどで自分を取り巻く偶然を楽しんだあとは、まじめなホロスコープで自分の星の運勢をズバリ当てちゃうのだ。



新刊

A5判 定価 1,200円

## X1ソフトプレス② トランプゲーム集

アクションゲームから思考ゲームまでトランプの各分野からピックアップしたゲームは8種類。マシン語による高速美麗カード表示ルーチンがゲームの興奮を盛り上げる。ゲーム自作派にもオススメの1冊。



新刊

A5判 定価 1,200円

■お求めは最寄りのマイコン・ショップ、書店。または郵送料を添えて下記へお申し込みください。

〒150 東京都渋谷区渋谷2-9-1 青山田中ビル TEL. 03(486)4500(代)

(株) エム・アイ・エー







GRAPHIC, SOUND, COMPILER

X1ハイテックファイル

# Another Remarkable Work

For X1, X1 Turbo Users



**MIA**

ISBN4-87170-044-5 C3055 ¥2200E

定価2,200円